

REPORT DOCUMENTATION PAGE			Form Approved OMB NO. 0704-0188		
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA, 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p>					
1. REPORT DATE (DD-MM-YYYY) 16-12-2014		2. REPORT TYPE Final Report		3. DATES COVERED (From - To) 1-May-2013 - 31-Mar-2019	
4. TITLE AND SUBTITLE Final Report: Science of Security Label - Scalability and Usability			5a. CONTRACT NUMBER W911NF-13-1-0154		
			5b. GRANT NUMBER		
			5c. PROGRAM ELEMENT NUMBER		
6. AUTHORS Scherlis, De Reno			5d. PROJECT NUMBER		
			5e. TASK NUMBER		
			5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAMES AND ADDRESSES Carnegie Mellon University 5000 Forbes Avenue Pittsburgh, PA 15213 -3815			8. PERFORMING ORGANIZATION REPORT NUMBER		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS (ES) U.S. Army Research Office P.O. Box 12211 Research Triangle Park, NC 27709-2211			10. SPONSOR/MONITOR'S ACRONYM(S) ARO		
			11. SPONSOR/MONITOR'S REPORT NUMBER(S) 63185-CS.89		
12. DISTRIBUTION AVAILABILITY STATEMENT Approved for Public Release; Distribution Unlimited					
13. SUPPLEMENTARY NOTES The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.					
14. ABSTRACT The Carnegie Mellon University School of Computer Science (SCS) proposes to perform research to accelerate the achievement of security and assurance goals for larger-scale software-reliant systems. Software-reliant systems are in increasingly critical roles throughout the DoD, manifesting capability in nearly every major functional area. Such systems may involve many separate components, rich and diverse supply chains, and complex interactions with human operators. Assuring security and quality in software-reliant systems is not only increasingly critical to operational success, but					
15. SUBJECT TERMS FPR: Final report					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON William Scherlis
a. REPORT UU	b. ABSTRACT UU	c. THIS PAGE UU			19b. TELEPHONE NUMBER 412-268-8741

Report Title

Final Report: Science of Security Lablet - Scalability and Usability

ABSTRACT

The Carnegie Mellon University School of Computer Science (SCS) proposes to perform research to accelerate the achievement of security and assurance goals for larger-scale software-reliant systems. Software-reliant systems are in increasingly critical roles throughout the DoD, manifesting capability in nearly every major functional area. Such systems may involve many separate components, rich and diverse supply chains, and complex interactions with human operators.

Assuring security and quality in software-reliant systems is not only increasingly critical to operational success, but it is also increasingly challenging due to the continued growth in complexity, scale, and criticality. Success in developing and evaluating critical and infrastructural systems demands high levels of sophistication in the technical aspects of cybersecurity, software and hardware design, and human- systems interaction. It also demands a strongly scientific attitude, recognizing the difficulties in the quest for effective means to evaluate and measure, exacerbated by the challenges of continual technological change.

Principal areas of focus. The proposed project focuses on advancing our ability to address these security and assurance challenges by focusing on two particular dimensions of the technical challenge of developing secure and assured systems. These are scalability and usability. We consider these from the standpoint of (a) the construction of new systems that are intended to be highly secure, (b) the evaluation of existing designs and systems with respect to security-related quality attributes, and (c) the sustainment and evolution of existing systems to achieve enhancements to security and quality attributes. A key features of the project is the advancement of a more explicitly scientific approach. Indeed, the overall research methodology and management strategy is driven by this latter consideration.

Enter List of papers submitted or published that acknowledge ARO support from the start of the project to the date of this printing. List the papers, including journal references, in the following categories:

(a) Papers published in peer-reviewed journals (N/A for none)

Received

Paper

08/14/2014 38.00 K. Joseph, G. P. Morgan, M. K. Martin, K. M. Carley. On the Coevolution of Stereotype, Culture, and Social Relationships: An Agent-Based Model, Social Science Computer Review, (12 2013): 0. doi: 10.1177/0894439313511388

TOTAL: 1

Number of Papers published in peer-reviewed journals:

(b) Papers published in non-peer-reviewed journals (N/A for none)

Received

Paper

TOTAL:

Number of Papers published in non peer-reviewed journals:

(c) Presentations

- 1) Arbob Ahmad and Robert Harper, "An Epistemic Formulation of Information Flow Analysis" - SoS Quarterly Lablet PI Meeting, Carnegie Mellon University, Pittsburgh, PA, July 1-2, 2014
- 2) Kathleen Carley, "Crisis Mapping: Big Data from a Dynamic Network Analytic Perspective" - World Summit on Big Data and Organization Design, Paris, France, May 16-17, 2013 - Invited Plenary
- 3) Kathleen Carley. "Geo-Spatial Network Analysis: Applications and Challenges" - 4th International Workshop on Location-Based Social Networks (LBSN 2012) at UBICOM, Pittsburgh, PA. September 8, 2012 - Invited Keynote
- 4) Kathleen Carley. "Dynamic Network Analysis: Security Applications" - Creighton University, Omaha Nebraska, 7/2013 - Invited Talk
- 5) Kathleen Carley. "Network-Centric Simulation and Virtual Experimentation" - 21st Behavioral Representation in Modeling and Simulation (BRiMS) Conference. Amelia Island, FL, March 12-15, 2012
- 6) Kathleen Carley. "Multi-Level Resilience Modeling: Evaluating Organizational Topologies for their Resilience to Attacks" - ONR sponsored Workshop on Social Cyber Security, Pittsburgh PA, 6/2013 - Poster Presentation
- 7) Kathleen M. Carley, "Analyzing and Simulating Dynamic Networks" - Rand Conference, Arlington, VA, March 2014
- 8) Kathleen M. Carley, "Dynamic Network Analytics for Cyber Warfare" - CENTCOM Technical Exchange, Tampa, FL, March 2014
- 9) Kathleen M. Carley, "Network Analysis and Visualization" - Approaches to Dynamic Network and Scientometric Analysis within the IC, Washington DC, November 2013
- 10) Kathleen M. Carley, "Networks and Agents: The Value of a Multi-Level Approach to Agent-Based Dynamic- Network Modeling" - Statistical and Applied Mathematical Sciences Institute (SAMSI), Raleigh-Durham, NC, August 2013
- 11) Kathleen M. Carley, "Dynamic Network Analysis: Security Applications" - Creighton University, Omaha Nebraska, July 2013
- 12) Geoffrey Morgan, "Construct" - CASOS Summer Institute, Carnegie Mellon University, Pittsburgh, PA, June 15-22, 2014
- 13) Kathleen Carley. "Crisis Mapping: Big Data from a Dynamic Network Analytic Perspective." - World Summit on Big Data and Organization Design, Paris, France, May 16-17, 2013 - Invited Plenary
- 14) Kathleen Carley. "Geo-Spatial Network Analysis: Applications and Challenges." - 4th International Workshop on Location-Based Social Networks (LBSN 2012) at UBICOM, Pittsburgh, PA. September 8, 2012 - Invited Keynote
- 15) Kathleen Carley. "Dynamic Network Analysis: Security Applications." - Creighton University, Omaha Nebraska, July 2013 - Invited Talk
- 16) Ghita Mezzour. "International Cyber Attack Network Analysis." - ONR sponsored Workshop on Social Cyber Security, Pittsburgh PA, 6/2013 - Poster Presentation
- 17) Kenneth Joseph and Kathleen Carley. "Group-based Constructuralism: modeling the evolution of groups, ties, culture, and cognition" - CASOS Summer Institute, Carnegie Mellon University, Pittsburgh, PA, June 15-22, 2014 - Poster Presentation
- 18) Ghita Mezzour and Kathleen Carley. "Putting Cyber-Attacks on the World Map" - CASOS Summer Institute, Carnegie Mellon University, Pittsburgh, PA, June 15-22, 2014 - Poster Presentation
- 19) Geoffrey Morgan and Kathleen Carley. "Organizational Resilience and the Meta-Network Formalism" - CASOS Summer Institute, Carnegie Mellon University, Pittsburgh, PA, June 15-22, 2014 - Poster Presentation
- 20) Geoffrey Morgan and Kathleen Carley. "Modeling Organizational Resiliency to Cyber-Attacks" - Carnegie Mellon University Lablet Meeting, Pittsburgh, PA, July 1-2, 2014 - Poster Presentation
- 21) Ghita Mezzour, Kathleen Carley, and Richard Carley. "Global Mapping of Cyber Attacks" - Carnegie Mellon University Lablet Meeting, Pittsburgh, PA, July 1-2, 2014 - Poster Presentation
- 22) Kathleen M. Carley, "Dynamic Network Analytics for Cyber Warfare" - CENTCOM Technical Exchange, Tampa, FL, March 2014
- 23) Kathleen M. Carley, "Network Analysis and Visualization" - Approaches to Dynamic Network and Scientometric Analysis within the IC, Washington DC, November 2013

- 24) Kathleen M. Carley, "Dynamic Network Analysis" - Soft Power Solutions, Chantilly VA, June 2014
- 25) Kathleen M. Carley, "Remote CBRNE Assessment using Dynamic Network Methods" - Defense Threat Reduction Agency, Washington DC, May 2014
- 26) Kathleen M. Carley, "A Global Perspective on Cyber Attacks" - NSA Security Lablet, Carnegie Mellon University, Pittsburgh, PA, September 2013
- 27) Kathleen M. Carley, "Dynamic Network Analysis: Security Applications" - Creighton University, Omaha Nebraska, July 2013 - Invited Talk
- 28) Jon Storrick, "Geo-Spatial Networks" - CASOS Summer Institute, Carnegie Mellon University, Pittsburgh, PA, June 15-22, 2014
- 29) Ghita Mezzour, "Nuclear, Bio, and Cyber Networks Social Influence Modeling" - CASOS Summer Institute, Carnegie Mellon University, Pittsburgh, PA, June 15-22, 2014
- 30) Geoff Morgan, "Validation Extended Construct" - CASOS Summer Institute, Carnegie Mellon University, Pittsburgh, PA, June 15-22, 2014
- 31) Geoff Morgan, "Resiliency Modeling" - CASOS Summer Institute, Carnegie Mellon University, Pittsburgh, PA, June 15-22, 2014
- 32) Wei Wei, "Geo-temporal Networks" - CASOS Summer Institute, Carnegie Mellon University, Pittsburgh, PA, June 15-22, 2014
- 33) Alain Forget, S. Komanduri, Alessandro Acquisti, Nicolas Christin, Lorrie Cranor, Rahul Telang. "Security Behavior Observatory: Infrastructure for Long-term Monitoring of Client Machines" - Symposium and Bootcamp on the Science of Security (HotSoS) 2014, ACM. Raleigh, NC, April 8-9, 2014
- 34) Alain Forget, Alessandro Acquisti, Nicolas Christin, Lorrie Cranor, Rahul Telang. "Deploying the Security Behavior Observatory: An Infrastructure for Long-term Monitoring of Client Machines" - NSA Science of Security Lablet Meeting, July 2014. Carnegie Mellon University, Pittsburgh, PA, July 1-2, 2014, - Poster Presentation
- 35) Alain Forget, Alessandro Acquisti, Lorrie Faith Cranor, Nicolas Christin, Rahul Telang. "Security Behavior Observatory. Lightning talk at the Symposium on Usable Privacy and Security" - ACM, July 2013, Newcastle, UK.
- 36) Alain Forget. "Flying South For The Career." - Invited talk at the ISSNNet 2013 Annual Workshop, NSERC, April 23-26, 2013, Victoria, Canada.
- 37) Alain Forget, Alessandro Acquisti, Lorrie Faith Cranor, Nicolas Christin, Rahul Telang. "Security Behavior Observatory." Lightning talk at the CyLab Usable Privacy and Lunch seminar, CMU, March 2013, Pittsburgh, USA.
- 38) Nathan Fulton, Cyrus Omar, and Jonathan Aldrich. "Statically Typed String Sanitation Inside a Python" - PSP 2014 : First International Workshop on Privacy and Security in Programming, Portland, OR, October 20-24, 2014
- 39) Darya Kurilova, Alex Potanin, and Jonathan Aldrich. "Wyvern: Impacting Software Security via Programming Language Design" - Workshop on Evaluation and Usability of Programming Languages and Tools (PLATEAU), 2014, SPLASH, Portland, OR, October 20-24, 2014
- 40) Michael Coblenz, Jonathan Aldrich, Brad Myers, and Joshua Sunshine. "Considering Productivity Effects of Explicit Type Declarations" - Workshop on Evaluation and Usability of Programming Languages and Tools (PLATEAU), 2014, SPLASH, Portland, OR, October 20-24, 2014
- 41) Jonathan Aldrich. "Extensible Languages" IFIP TC2 working group on programming language design (#2.16), Aarhus, Denmark, August 21-25, 2013
- 42) Cyrus Omar, Benjamin Chung, Darya Kurilova, Alex Potanin, and Jonathan Aldrich. "Type Directed, Whitespace-Delimited Parsing for Embedded DSLs". - Presentation at First Workshop on Domain Specific Languages Design and Implementation (DSLDI), 2013, Montpellier, France, July 1, 2013
- 43) Jonathan Aldrich. "Architectural Control" Presentation at the IFIP Working Group on Language Design, Portland, OR, June 6, 2014

- 44) Sam Malek. "Toward the Making of Software that Learns to Manage Itself" Keynote at the 27th Brazilian Symposium on Software Engineering (SBES 2013). Brasilia, Brazil, Sept 29 - October 4, 2013.
- 45) Sam Malek. "Automated Security Testing of Mobile Applications" FedMobileCamp hosted by NGA/InnoVision, Reston, VA, August 2013.
- 46) Sam Malek and Marija Mikic-Rakic. "A Framework for Improving a Distributed Software System's Deployment Architecture" Delft University of Technology, Delft, Netherlands, June 2013
- 47) David Garlan, Mary Shaw. "Software Architecture: Reflections on an Evolving Discipline" - International Conference on Software Engineering and Applications (ICSE), San Francisco, CA, May 18-26, 2013, Keynote
- 48) David Garlan. "Software Architecture: Perspectives and Challenges" - 7/2014 Beihang University
- 49) David Garlan. "Self-Healing Systems" - 2/2014 Samsung Electronics
- 50) Andre Platzer. "Logical Foundations of Cyber-Physical Systems" - Invited talk at High Confidence Software and Systems Conference, 2014 (HCSS'14), Annapolis, MD, May 20, 2014
- 51) Andre Platzer. "Foundations of Cyber-Physical Systems" - Invited course at MAP-i, Universities of Minho, Braga, Porto and Aveiro, Portugal, March 2014
- 52) Andre Platzer. "Logical Foundations of Cyber-Physical Systems" - NSF Workshop for Aspiring PIs in Cyber-Physical Systems, Washington, DC, February 18-19, 2014
- 53) Andre Platzer. "Developing a Successful NSF Proposal" - NSF Workshop for Aspiring PIs in Cyber-Physical Systems, Washington, DC, February 18-19, 2014
- 54) Andre Platzer. "Logic of Dynamical Systems" - Invited Research School at École Normale Supérieure (ENS) de Lyon, France, January 2014
- 55) Andre Platzer. "Hybrid Systems Verification" - Invited talk at 4th Workshop Formal Methods for Robotics and Automation, Berlin, Germany, June 27, 2013
- 56) Andre Platzer. "How to Explain Cyber-Physical Systems to Your Verifier" - Invited talk at 5th Working Conference on Verified Software: Theories, Tools, and Experiments (VSTTE'13), Atherton, CA, May 17-19, 2013
- 57) Andre Platzer. "Logic of Hybrid Games" - Invited talk at LCCC Focus Period and Workshop on Formal Verification of Embedded Control Systems, Lund, Sweden, April, 2013
- 58) Limin Jia. "Proving Trace Properties of Programs that Execute Adversary-supplied Code" - Presented at PLClub at University of Pennsylvania, June 7, 2013
- 59) Limin Jia. "Proving Trace Properties of Programs that Execute Adversary-supplied Code" - Presented at Seminar at Princeton University, June 20th, 2013
- 60) Limin Jia. "Proving Trace Properties of Programs that Execute Adversary-supplied Code" - INRIA, Paris Rocquencourt, June 5, 2014
- 61) Frank Pfenning. "Concurrent Programming in Linear Type Theory" Workshop on the Mathematical Structures of Computation, Lyon, France, February 2014
- 62) Frank Pfenning. "Linear Logic and Session Types" 4-lecture course at the BETTY Summer School on Behavioral Types, Lovran, Croatia, June/July 2014
- 63) Ju-Sung Lee, Juergen Pfeffer. "Measurement Accuracy in Samples of Online Communication Networks" - EUSN, 1st European Conference on Social Networks, Barcelona, Spain, July 1-4 2014
- 64) Ju-Sung Lee, Juergen Pfeffer. "Robustness of Network Metrics in the Context of Cyber Communication Data" - Quarterly Lablet meeting, Carnegie Mellon University, Pittsburgh, PA, July 1-2 2014. - Poster Presentation
- 65) Ju-Sung Lee, Jürgen Pfeffer. "Approximating Network Measures for Large Scale Networks of Varying Size, Density, Typologies, and Subsampling Levels" - Sunbelt Network Analysis Conference, March 19-23, 2014

66) Jürgen Pfeffer, Kathleen M. Carley, Bradley Schmerl, David Garlan. "Network Analysis for Big Data in Resource-Constrained Environments" - Sunbelt Network Analysis Conference 2013, Hamburg, Germany, May 21-26, 2013

67) Jürgen Pfeffer. "Composability of Big Data and Algorithms for Social Networks Analysis Metrics" - Quarterly Lablet meeting at Urbana-Champaign, IL, May 1, 2014

68) Jürgen Pfeffer. "Describing Structural Change in Networked Systems" - Quarterly Lablet meeting at Urbana-Champaign, IL, May 1, 2014

Number of Presentations: 68.00

Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

<u>Received</u>	<u>Paper</u>
-----------------	--------------

TOTAL:

Number of Non Peer-Reviewed Conference Proceeding publications (other than abstracts):**Peer-Reviewed Conference Proceeding publications (other than abstracts):**ReceivedPaper

- 08/05/2014 1.00 S. Komanduri, , Alessandro Acquisti, , Nicolas Christin, , Lorrie Cranor, , Robert Telang, Alain Forget, . Building the Security Behavior Observatory: An Infrastructure for Long-term Monitoring of Client Machines, Symposium and Bootcamp on the Science of Security (HotSoS) 2014, ACM. 08-APR-14, . : ,
- 08/08/2014 15.00 Cyrus Omar, Darya Kurilova, Ligia Nistor, Benjamin Chung, Alex Potanin, Jonathan Aldrich. Safely Composable Type-Specific Languages, Proc. European Conference on Object-Oriented Programming, 2014. 28-JUL-14, . : ,
- 08/08/2014 20.00 Filipe Militão, Jonathan Aldrich, Luís Caires. Substructural Typestates, In Programming Languages meets Program Verification, 2014. 21-JAN-14, . : ,
- 08/08/2014 19.00 Ligia Nistor, Jonathan Aldrich, Stephanie Balzer, Hannes Mehnert. Object Propositions, In Formal Methods, 2014. 12-MAY-14, . : ,
- 08/08/2014 18.00 Michael Maass, Jonathan Aldrich, William Scherlis. In-Nimbo Sandboxing, Proc. Science of Security (HotSOS), 2014. 08-APR-14, . : ,
- 08/08/2014 17.00 Filipe Militão, Jonathan Aldrich, Luís Caires. . Rely-Guarantee Protocols, Proc. European Conference on Object-Oriented Programming, 2014. 28-JUL-14, . : ,
- 08/08/2014 16.00 Joshua Sunshine, James Herbsleb, Jonathan Aldrich. Structuring Documentation to Support State Search: A Laboratory Experiment about Protocol Programming, Proc. European Conference on Object-Oriented Programming, 2014. 28-JUL-14, . : ,
- 08/14/2014 30.00 Tingting Yu, Witawas Srisa-an, Gregg Rothermel. SimRT: An Automated Framework to SupportRegression Testing for Data Races, International Conference on Software Engineering (ICSE) 2014. 31-MAY-14, . : ,
- 08/14/2014 41.00 David Garlan. Software architecture: a travelogue, Future of Software Engineering. 31-MAY-14, Hyderabad, India. : ,
- 08/14/2014 40.00 Javier Cámara, Gabriel A. Moreno, David Garlan. Stochastic Game Analysis and Latency Awarenessfor Proactive Self-Adaptation, 9th International Symposium on Software Engineering for Adaptive and Self-Managing Systems. 02-JUN-14, . : ,
- 08/14/2014 39.00 Bradley Schmerl, Javier Camara, Jeffrey Gennari, David Garlan,, Paulo Casanova, Gabriel A. Moreno, Thomas J. Glazier, Jeffrey M. Barnes. Architecture-Based Self-Protection: Composing and Reasoning about Denial-of-Service Mitigations, HotSoS 2014: 2014 Symposium and Bootcamp on the Science of Security. 08-APR-14, . : ,
- 08/16/2014 48.00 Cyrus Omar, Darya Kurilova, Ligia Nistor, Benjamin Chung, Alex Potanin, Jonathan Aldrich. Safely Composable Type-Specific Languages , The European Conference on Object-Oriented Programming, 2014.. 28-JUL-14, . : ,
- 08/16/2014 50.00 Alireza Sadeghi, Naeem Esfahani, Sam Malek. Mining the Categorized Software Repositories toImprove the Analysis of Security Vulnerabilities, 17th International Conference on Fundamental Approaches to Software Engineering (FASE 2014), Grenoble, France. 05-APR-14, . : ,

- 08/16/2014 49.00 Filipe Militao, Jonathan Aldrich, Lu's Caires. Rely-Guarantee Protocols, The European Conference on Object-Oriented Programming, 2014.. 28-JUL-14, . : ,
- 08/18/2014 51.00 Eric Yuan, Naeem Esfahani, Sam Malek. Automated Mining of Software Component Interactions for Self-Adaptation, 9th International Symposium on Software Engineering for Adaptive and Self-Managing Systems. 03-JUN-14, . : ,
- 08/18/2014 53.00 Jonathan Aldrich, Cyrus Omar, Alex Potanin, Du Li. Language-Based Architectural Control, International Workshop on Aliasing, Capabilities, and Ownership (IWACO '14),. 29-JUL-14, . : ,
- 08/18/2014 52.00 Paulo Casanova, David Garlan, Bradley Schmerl, Rui Abreu. Diagnosing Unobserved Components in Self-Adaptive Systems, 9th International Symposium on Software Engineering for Adaptive and Self-Managing Systems. 02-JUN-14, . : ,
- 12/09/2014 70.00 Marwan Abi-Antoun, Radu Vanciu . Finding Architectural Flaws using Constraints, IEEE/ACM Conference on Automated Software Engineering (ASE), 2013. Palo Alto, CA, November 11, 2013. 11-NOV-13, . : ,
- 12/09/2014 62.00 Nathan Fulton. Domain Specific Security through Extensible Type Systems, Proceedings of the SPLASH Student Research Competition, 2012. Tucson, AR, October 19-26, 2012. 19-OCT-12, . : ,
- 12/09/2014 71.00 Sam Malek, , Bradley Schmerl, , David Garlan, , Jeff Gennari, Eric Yuan, . Architecture-Based Self-Protecting Software Systems, In Proceedings of the Ninth International ACM Sigsoft Conference on the Quality of Software Architectures (QoSA 2013), Vancouver, BC, Canada, June 17-21, 2013. 17-JUN-13, . : ,
- 12/09/2014 63.00 Jonathan Aldrich. The Power of Interoperability: Why Objects Are Inevitable, Proceeding Onward! 2013 Proceedings of the 2013 ACM international symposium on New ideas, new paradigms, and reflections on programming & software. . : ,
- 12/09/2014 66.00 Marwan Abi-Antoun , Sumukhi Chandrashekar , Radu Vanciu , Andrew Giang. Are Object Graphs Extracted Using Abstract Interpretation Significantly Different from the Code, IEEE International Working Conference on Source Code Analysis and Manipulation (SCAM), 2014. Victoria, BC, Canada, September 28-29, 2014. 28-SEP-14, . : ,
- 12/09/2014 67.00 Radu Vanciu, Ebrahim Khalaj, Marwan Abi-Antoun. Comparative Evaluation of Architectural and Code-Level Approaches for Finding Security Vulnerabilities., Workshop on Security Information Workers, co-located with the ACM Conference on Computer and Communications Security (CCS), 2014. Scottsdale, AZ, November 7, 2014. 07-NOV-14, . : ,
- 12/09/2014 68.00 Ebrahim Khalaj , Radu Vanciu , Marwan Abi-Antoun. Is There Value in Reasoning about Security at the Architectural Level: a Comparative Evaluation?, Symposium and Bootcamp on the Science of Security (HotSoS), 2014. Raleigh, NC, April 8-9, 2014 - Poster Presentation. 08-APR-14, . : ,
- 12/09/2014 58.00 Blase Ur, , Patrick Gage Kelley, , Saranga Komanduri, , Joel Lee, , Michael Maass, , Michelle Mazurek, , Timothy Passaro, , Richard Shay, , Timothy Vidas, , Lujo Bauer, , Nicolas Christin, , Lorrie Faith Cranor. How Does Your Password Measure Up? The Effect of Strength Meters on Password Creation, Proceedings of 21st USENIX Security Symposium, Bellevue, WA, August 8-10, 2012 . 08-AUG-12, . : ,
- 12/09/2014 59.00 Jonathan Aldrich, Cyrus Omar, , Benjamin Chung, , Darya Kurilova, , Alex Potanin. Type Directed, Whitespace-Delimited Parsing for Embedded DSLs, Proceedings of International Workshop on Globalization of Domain Specific Languages (GlobalDSL), 2013. Montpellier, France, July 2, 2013. 02-JUL-13, . : ,

12/09/2014	60.00	Ligia Nistor, , Darya Kurilova, , Stephanie Balzer, , Benjamin Chung, , Alex Potanin, , Jonathan Aldrich. Wyvern: A Simple, Typed, and Pure Object-Oriented Language, Proceedings of Workshop on Mechanisms for Specialization, Generalization, and Inheritance (MASPEGHI), 2013. Montpellier, France, July 1, 2013. 01-JUL-13, . : ,
12/09/2014	61.00	Simin Chen. Declarative Access Policies based on Objects, Relationships, and States, Proceedings of the SPLASH Student Research Competition, 2012. Tucson, AR, October 19-26, 2012. 19-OCT-12, . : ,
12/10/2014	77.00	Alexander H. Levis , Bahram Yousefi. Multi-Formalism Modeling for Evaluating the Effects of Cyber Exploits, Proc. 28thEuropean Conference on Modeling and Simulation (ECMS2014), Brescia, Italy, May 27-30, 2014; Springer, Heidelberg, Germany, 2014. 27-MAY-14, . : ,
12/10/2014	80.00	Alexander H. Levis , Bahram Yousefi. Multi-Formalism modeling for evaluating the effect of cyber exploits, 28th European Conference on Modeling and Simulation, Brescia, Italy, May 27-30, 2014.. 27-MAY-14, . : ,
TOTAL:		30

Number of Peer-Reviewed Conference Proceeding publications (other than abstracts):

(d) Manuscripts

<u>Received</u>	<u>Paper</u>
08/19/2014 32.00	Witawas Srisa-an. ReflexScope: A Hybrid Analysis Approach to Demystify Reflection Usage in Android Apps, (06 2014)
08/19/2014 33.00	Witawas Srisa-an. RaceDr: A Just-in-time Atomicity Violation Repair Framework, (06 2014)
08/19/2014 34.00	Witawas Srisa-an. Leverage Redundancy in Hardware Transactional Memory to Improve System Reliability, (06 2014)
TOTAL:	3

Number of Manuscripts:

Books

Received Book

TOTAL:

Received Book Chapter

12/10/2014 75.00 Geoffrey P. Morgan,, Kathleen M. Carley. Modeling Formal and Informal Ties within an Organization: A Multiple Model Integration, unknown: Emerald Group Publishing Limited, (01 2012)

TOTAL: 1

Patents Submitted

Michael Maass, Bill Scherlis, Jonathan Aldrich - Language and Framework for Development of Secure Mobile Apps
~~"In Nimbo Sandboxing"~~

Patents Awarded

Awards

Kathleen Carley - Learned Resiliency in Multi-Level Systems

-
- 1) Kathleen Carley - Member of the NAS/NRC Committee on Digital Math Library, 2012-13
 - 2) Kathleen Carley - DHS Homeland Security Science and Technology Advisory Committee, HSSTAC, SGE. Also member of special sub-group on cyber-security, 2012-13
 - 3) Kathleen Carley - IEEE Fellow, 2013
 - 4) Kathleen Carley - Allen Newell Award for Research Excellence - "For the creation of empirical methods to rigorously establish the impact of human communication on software quality." 2014
 - 5) Kathleen Carley - Member of the NAS ARL Review Panel, 2014

Kathleen Carley, Geo-Temporal Characterizations

- 1) Kathleen Carley - Member of the NAS/NRC Committee on Digital Math Library, 2012-13
- 2) Kathleen Carley, DHS Homeland Security Science and Technology Advisory Committee, HSSTAC, SGE , Also member of special sub-group on cyber-security, 2012-13

Jonathan Aldrich, A Language and Framework for Development of Secure Mobile Apps

- 1) Nathan Fulton, Cyrus Omar, and Jonathan Aldrich. "Statically Typed String Sanitation Inside a Python," PSP Best paper award
- 2) Cyrus Omar, Darya Kurilova, Ligia Nistor, Benjamin Chung, Alex Potanin, and Jonathan Aldrich. "Safely Composable Type-Specific Languages" awarded ECOOP 2014 Distinguished Paper Award

David Garlan, Jonathan Aldrich, Bradley Schmerl - Science of Secure Frameworks

- 1) Sam Malek, Mason Emerging Researcher/Scholar/Creator Award
- 2) David Garlan, ACM Fellow

Graduate Students

<u>NAME</u>	<u>PERCENT SUPPORTED</u>	Discipline
Arbob Ahmad	0.93	
Michael Arntzenius	0.29	
Vishal Dwivedi	0.04	
Hanan Hibshi	0.12	
Kuen-Bang Hou	0.60	
Krutika Kamilla	0.03	
Darya Kurilova	0.89	
Momin Malik	0.07	
Ghita Mezzour	0.49	
Geoffrey Morgan	0.61	
Cyrus Omar	0.37	
Jigan Patel	0.08	
Ashwini Giridhar Rao	0.36	
Cassandra Urmano	0.09	
Wei Wei	0.10	
Erik Peter Zawadzki	0.92	
FTE Equivalent:	5.99	
Total Number:	16	

Names of Post Doctorates

<u>NAME</u>	<u>PERCENT SUPPORTED</u>
Stephanie Balzer	0.47
Nikou Gunnemann-Gholizadeh	0.99
Du Li	0.84
Alex Potanin	0.10
New Entry	0.00
FTE Equivalent:	2.40
Total Number:	5

Names of Faculty Supported

<u>NAME</u>	<u>PERCENT SUPPORTED</u>	National Academy Member
Alessandro Acquisti	0.10	
Jonathan Aldrich	0.14	
Travis Breaux	0.20	
Kathleen Carley	0.17	
Nicolas Christin	0.08	
Lorrie Cranor	0.09	
Anupam Datta	0.09	
David Garlan	0.01	
Juergen Pfeffer	0.06	
Frank Pfenning	0.15	
Andrew Platzner	0.05	
William Scherlis	0.17	
Bradley Schmerl	0.16	
FTE Equivalent:	1.47	
Total Number:	13	

Names of Under Graduate students supported

NAME	PERCENT SUPPORTED	Discipline
Benjamin Chung	0.93	
Jia Jun Brandon Lum	0.37	
Rahul Manne	0.38	
Wen Jay Tan	0.59	
FTE Equivalent:	2.27	
Total Number:	4	

Student Metrics

This section only applies to graduating undergraduates supported by this agreement in this reporting period

The number of undergraduates funded by this agreement who graduated during this period:

The number of undergraduates funded by this agreement who graduated during this period with a degree in science, mathematics, engineering, or technology fields:.....

The number of undergraduates funded by your agreement who graduated during this period and will continue to pursue a graduate or Ph.D. degree in science, mathematics, engineering, or technology fields:.....

Number of graduating undergraduates who achieved a 3.5 GPA to 4.0 (4.0 max scale):.....

Number of graduating undergraduates funded by a DoD funded Center of Excellence grant for Education, Research and Engineering:.....

The number of undergraduates funded by your agreement who graduated during this period and intend to work for the Department of Defense

The number of undergraduates funded by your agreement who graduated during this period and will receive scholarships or fellowships for further studies in science, mathematics, engineering or technology fields:.....

Names of Personnel receiving masters degrees

NAME
Joshua Sunshine, December 2013
Michael Lanham, exp May 2015
Total Number:

2

Names of personnel receiving PHDs

NAME
Joshua Sunshine, December 2014
Michael Lanham, exp May 2015
Naeem Esfahaini, August 2014
Radu Vanciu, May 2014
Tingting Tu, August 2014
Total Number:

5

Names of other research staff

<u>NAME</u>	<u>PERCENT SUPPORTED</u>
Michael Kowalchuck	0.18
Limin Jia	0.04
Ju-Sung Lee	0.10
Jeremy Thomas	0.25
Amit Vasudevan	0.14
Alain Forget	0.10
FTE Equivalent:	0.81
Total Number:	6

Sub Contractors (DD882)

1 a. Sponsored Programs, University of Nebraska-Lincoln

1 b. 312 N. 14th St, Alexander West

Lincoln NE 68588

Sub Contractor Numbers (c): 1130163-310993

Patent Clause Number (d-1):

Patent Date (d-2):

Work Description (e): Pis: Matthew B. Dwyer and Witawas Srisa-anTask 1:The main objective is to detect race-

Sub Contract Award Date (f-1): 4/30/13 12:00AM

Sub Contract Est Completion Date(f-2): 9/30/14 12:00AM

1 a. Wayne State University

1 b. 5057 Woodward Ave.

13th Floor

Detroit MI 482024050

Sub Contractor Numbers (c): 1130163-311858

Patent Clause Number (d-1):

Patent Date (d-2):

Work Description (e): Proposed Tasks •Design a security policy specification language that can express fine-gra

Sub Contract Award Date (f-1): 5/1/13 12:00AM

Sub Contract Est Completion Date(f-2): 9/30/14 12:00AM

1 a. Wayne State University

1 b. Office of Research & Sponsored

Program Services

Detroit MI 482023692

Sub Contractor Numbers (c): 1130163-311858

Patent Clause Number (d-1):

Patent Date (d-2):

Work Description (e): Proposed Tasks •Design a security policy specification language that can express fine-gra

Sub Contract Award Date (f-1): 5/1/13 12:00AM

Sub Contract Est Completion Date(f-2): 9/30/14 12:00AM

1 a. University of Texas at San Antonio

1 b. One UTSA Circle

San Antonio TX 782490603

Sub Contractor Numbers (c): 1130163-311712

Patent Clause Number (d-1):

Patent Date (d-2):

Work Description (e): Dr. Jianwei Niu, University of Texas at San AntonioDr. Niu will collaborate with Dr. Tra

Sub Contract Award Date (f-1): 5/1/13 12:00AM

Sub Contract Est Completion Date(f-2): 9/30/14 12:00AM

1 a. University of Texas at San Antonio

1 b. 6900 North Loop 1604 West

San Antonio TX 782491130

Sub Contractor Numbers (c): 1130163-311712

Patent Clause Number (d-1):

Patent Date (d-2):

Work Description (e): Dr. Jianwei Niu, University of Texas at San AntonioDr. Niu will collaborate with Dr. Tra

Sub Contract Award Date (f-1): 5/1/13 12:00AM

Sub Contract Est Completion Date(f-2): 9/30/14 12:00AM

1 a. George Mason University

1 b. 4400 University Drive, MS 4C6

Fairfax VA 220304422

Sub Contractor Numbers (c): 1130163-311571

Patent Clause Number (d-1):

Patent Date (d-2):

Work Description (e): PI: Prof. Sam Malek (Faculty)Proposed ResearchJust like the application logic, the self-

Sub Contract Award Date (f-1): 5/1/13 12:00AM

Sub Contract Est Completion Date(f-2): 9/30/14 12:00AM

1 a. George Mason University

1 b. 4400 University Drive, MSN 4C6

Fairfax VA 220304422

Sub Contractor Numbers (c): 1130163-311571

Patent Clause Number (d-1):

Patent Date (d-2):

Work Description (e): PI: Prof. Sam Malek (Faculty)Proposed ResearchJust like the application logic, the self-

Sub Contract Award Date (f-1): 5/1/13 12:00AM

Sub Contract Est Completion Date(f-2): 9/30/14 12:00AM

1 a. University of Pittsburgh

1 b. 123 University Place

Lower Lobby

Pittsburgh PA 152132303

Sub Contractor Numbers (c): 1130163-311633

Patent Clause Number (d-1):

Patent Date (d-2):

Work Description (e): PI-Scott BeachStatement of WorkUCSUR will perform the following tasks for the R-USI

Sub Contract Award Date (f-1): 5/1/13 12:00AM

Sub Contract Est Completion Date(f-2): 9/30/14 12:00AM

1 a. University of Pittsburgh

1 b. 350 Thackeray Hall

139 University Place

Pittsburgh

PA

152602600

Sub Contractor Numbers (c): 1130163-311633

Patent Clause Number (d-1):

Patent Date (d-2):

Work Description (e): PI-Scott BeachStatement of WorkUCSUR will perform the following tasks for the R-USI

Sub Contract Award Date (f-1): 5/1/13 12:00AM

Sub Contract Est Completion Date(f-2): 9/30/14 12:00AM

Inventions (DD882)

Scientific Progress

Jonathan Aldrich - Race Vulnerability Study and Hybrid Race Detection

- We finished the implementation to adapt the results of static analyses with dynamic analyses by feeding static analysis to the JVM when it launches. This increases the start-up time of JVM, but this overhead pays off for long-running applications.
- We created a database of race-related vulnerabilities and associated artifacts. The database now includes race-related vulnerabilities from the four major browsers (IE, Chrome, Safari, and Firefox). Vulnerabilities were culled from the National Vulnerability database, project bug repositories, and security mailing lists (e.g. Full Disclosure). For open source browsers, Chrome and Firefox, tests to replicate vulnerabilities and patches to fix vulnerabilities are included in the database.
- We formalized our race detection algorithm. The formalization consists of static and dynamic semantics that are consistent with the program analyses we implemented in our hybrid race detection system.
- We are working to develop dynamic analysis techniques inside Just-In-Time compilers. As static analysis tools become more effective at detecting security vulnerabilities, malware writers have to rely more on dynamism of today's software to deliver malicious intents. For example, the use of custom class loaders to dynamically load malicious code segments or the use of reflection to invoke external malicious methods provides ways for malware to escape detection efforts by static analysis tools. As code segments are being loaded through these mechanisms, we must be able to inspect them for potential malicious intents. They propose an approach to use a JIT compiler to detect dynamically loaded code for potential vulnerabilities during deployment. We are currently modifying the JIT compiler in OpenJDK to detect information leakage. As of now, the work is about 65% complete.
- We are also developing dynamic taint analysis framework to monitor information flow during deployment. Activity 1 provides us with the capability to detect potential vulnerabilities in dynamically loaded classes. To verify whether these vulnerabilities are real, we need to observe information that is flowing into methods of these classes. For example, if our approach detects that a method can store information to the SD card, it flags this method as containing a potential vulnerability. The real vulnerability can be confirmed by observing if sensitive information, such as entries from the address book, can actually reach this method. To do so, we need to be able to perform dynamic taint analysis. We are currently building the dynamic taint analysis framework in OpenJDK. The work is about 70% complete.

Jonathan Aldrich - A Language and Framework for Development of Secure Mobile Applications

Activity in progress:

- We developed a formal system for safely composing separately defined type system fragments with modular type constructors. We establish several strong semantic guarantees, notably type safety, stability of typing under extension and conservativity: that the type invariants that a finite set of fragments maintain are conserved under extension.
- The implementation of the Wyvern language continued. We added several important language features (e.g. type parameters), improved interoperability with Java, and completed a first implementation of language extensibility.
- We continued to define Wyvern in its specification. The working draft is available at <http://www.cs.cmu.edu/~aldrich/securemobileweb/spec-rationale.html>

Major results:

- In-Nimbo Sandboxing (HotSOS '14). In-Nimbo Sandboxing is a new concept that encapsulates untrusted or hard-to-assure computation by running it on ephemeral computing resources in a cloud computing environment. It has the advantage that any malicious state resulting from the computation will not persist in a high-value environment even if other controls on the computation fail. In-Nimbo Sandboxing can provide additional security in a mobile, desktop, or web environment. As a proof of concept, we built a sandbox for Adobe Reader, which has been repeatedly compromised in the past. We also developed a novel scheme for analyzing the level of security provided by a sandbox, enabling sandboxing techniques to be compared effectively.
- Safely Composable Type-Specific Languages (ECOOP '14). Command injection vulnerabilities are common in part because programmers compose commands by combining strings rather than using more structured, but inconvenient, representations such as prepared SQL statements. A promising mitigation is to provide programmers with mechanisms for constructing commands that are as convenient as strings while being as secure as prepared SQL statements. Mechanisms for embedding domain-specific languages (DSLs) for constructing commands within a programming language exist, but none have achieved widespread use, in part because prior techniques were unmodular, so that separately-defined embedded DSLs could not be used together. We describe a novel mechanism called type-specific languages that supports modular DSL embeddings by

associating a unique DSL with appropriate types. When the programmer wants to construct an instance of such a type--a database query, for example--he or she can define the instance using the associated DSL (SQL, in this example). Our paper describes the mechanism and states safety and composability properties of the design. The paper also includes results from an empirical study suggesting that strings are in very widespread use for constructing domain-specific object structures, indicating that our approach should have broad applicability.

•Structuring Documentation to Support State Search: A Laboratory Experiment about Protocol Programming (ECOOP '14). Incorrect use of libraries that require developers to follow a protocol is at the core of significant security vulnerabilities, such as the SSL libraries described in the CCS'12 paper "The most dangerous code in the world: validating SSL certificates in non-browser software." We developed a novel tool-supported intervention that makes the states defined in a library more visible in automatically generated documentation such as Javadoc. We then performed a quantitative study to find out whether this approach could help programmers use state-based libraries more quickly and with fewer errors. We found that the intervention makes programmers 2 times faster at performing state-related tasks. Furthermore, programmers using the documentation intervention made 7 times fewer mistakes in answering questions about the code they were writing. Overall, our results contribute to the science of security by demonstrating that better documentation of library code can assist programmers in writing code more quickly and accurately in the context of state-based libraries such as SSL. The paper also provides one of the first empirical results that directly supports the productivity benefits of putting security-related design intent into code.

David Garlan, Bradley Schmerl, Jonathan Aldrich - Science of Secure Frameworks

•We have almost completed a model of probe placement, reporting on the foundational theory at SEAMS 2014 that deals with a formalism and proofs for determining upper and lower bounds on required probes. Building on this work will allow us to reduce potential vulnerabilities that could be introduced by the monitoring required for self-adaptation by allowing us to reason about the minimal set of required probes to be able to diagnose issues in the system (including security issues).

•We refined our scientific approach to mitigating CSRF attacks based on architectural design intent relating to web interaction protocols. Two conference papers were accepted, and while they were supported primarily via other sources, they will contribute to the Science of Secure Frameworks project. [Omar et al. 2014], supported by the sister lablet project "A Language and Framework for Development of Secure Mobile Applications," describes language extensibility mechanisms that can be used by frameworks to provide secure extension points to their plugins. [Militao et al. 2014] describes an approach to specifying and verifying interaction protocols that we plan to leverage in mitigating CSRF vulnerabilities.

•Developed a tool called ARMOUR that leverages novel data mining techniques to detect security attacks from the interactions that arise in the system's runtime architecture. One of the obstacles with using a tool such as ARMOUR to detect anomalous behavior in arbitrary systems is that there is no universal optimal value for the underlying mining algorithms' parameters. That is, for each system, parameters such as support and confidence have to be carefully selected to effectively detect anomalous behavior. GMU team developed a novel approach for the selection of parameters based on the behavioral characteristics of the system to minimize the false positive and negative rates. The approach automatically selects and tunes the parameters based on the observed historical variability in the system and its use cases.

•Developed and tested an approach for detecting data flow vulnerabilities, called Scoria, that uses static analysis to determine dataflow between components in the system. This was tested on over 30 test cases from the DroidBench benchmark, and extended the benchmark with additional test cases.

Travis Breaux - Usability and Secure Requirements

In progress: security patterns produced from expert knowledge by applying Situation Awareness, which is a technique for eliciting experts' descriptive and prospective thoughts, to security problems.

In progress: a security pattern construction protocol based on the requirements inquiry-cycle model, which allows security analysts to incrementally update a pattern to address emerging security challenges.

In progress: a security pattern catalogue based on attributes extracted from security standards using text analysis and machine learning.

Kathleen M. Carley - Geo-Temporal Characterization of Security Threats

****See Attachments for graphs**

Objective

The objective of this project is to empirically characterize the nature of the current threat environment and to test a series of existing hypotheses about that threat environment using Symantec data. Our focus is global. The basic theory is that the potential severity of the threat is a function of the political environment rather than the technology. Questions to be addressed empirically include:

4. What is the likelihood of a catastrophic threat? Hypothesis: Most attacks are small.

5. How does the likelihood differ by type of threat? Hypothesis: there are no differences by type of threat.

6. Do these answers differ by country? Hypothesis A: Once GDP, internet penetration, and the number of attacks are accounted for there are no differences by country. Hypothesis B: The likelihood of a company being attacked depends on their position in the alliance/enmity network.

Background

In their 2011 survey Symantec found that the number one cyber risk business concern was external cyber-attacks, followed by concerns about both unintentional insider error (2nd risk) and intentional insider error (3rd risk) (Symantec, 2011, pg 9).

Analysis by Verizon's cyber forensics team indicates that the massive increase in external threats overshadows insider attacks (Verizon, 2012b, pg 21). See also Richardson (2008). Despite the increase in external threats little is known about the source of such threats; or the global implications this evolving threat environment.

Wynne (2010) notes that the need to do attribution and forensics is critical to stem the tide of cyber-attacks. To meet this need, an understanding of the threat environment at the global level is needed. Cyber security, at the global level, is critical on a number of fronts including countering terrorism (Westby, 2007). At the global level, cyber security requires not only attribution and forensics, but harmonized laws and effective information sharing. In spite of this growing consensus there is still little empirical understanding of the global cyber threat environment, an understanding that is critical for forensics. We have found that most global information sharing is done through security providers and the data is only now becoming available and in restricted form given the dual privacy and security needs that must be met.

We note that reliance on anecdotal evidence can be damaging for the Science of Security. As empirical findings come to light, assumptions about the nature of cyber threat are changing. For example, in 2004, Byre and Lowe showed that process control and SCADA systems were not immune to attack using incidence data. Further basic assumptions are falling as new empirical evidence comes to light. This is creating a new baseline against which forensics can operate. While there are an increasing body of findings focused on specific threats and empirical assessments of key incidents; there is less understanding of the human socio-behavioral factors, particularly at the global level.

At the global level, multiple conceptual frameworks abound. For example, Kshetri (2005) argues that country level differences in the regulative, normative and cognitive legitimacy of different types of web attacks lead to differences in the extent to which organized crime can use the internet in those countries. BSA (2010) provides guiding principles for global cyber security. Broadhurst (2006) argues that cyber-attacks are traditional crimes in a new venue which makes traditional forensic methods obsolete. And so on. Despite the recognition of the global nature of the cyber threat there is little characterization of that threat.

Approach

The basic approach used was to develop code for extracting country level indicators of attacks and attack paths from the Symantec data. The analysis is based on the Symantec WINE telemetry data set. WINE is a platform through which external researchers can access data sets used within Symantec Research labs. To the best of our knowledge, Symantec is currently the only security company that makes such platform available to external researchers. The telemetry data set consists of attack reports from more than 10 million Symantec customer computers worldwide.

This was then combined with other open-source data to create a global threat profile. An example of such additional open source data is the ICT index circa 2010 for all countries. The ICT index is a combined measure based on 11 indicators including adult literacy, internet access and so on. The data was fused to create a global indicators data set and was then assessed using network analytics and standard statistical procedures. Based on this data a network based model of the impact of hostilities and other factors on geo-cyber-attack-network was developed.

Summary of Key Results

There are two key aspects to this research. The first is the characterization of the threat profile. The second is the assessment of the over-all global threat with respect to cyber attacks.

In terms of the general threat profile, our results led to an empirical characterization of the change in cyber-threat over the past several years. This work indicated that web attacks account for the vast majority of attacks in the IPS catalog. Around 2003, worms and viruses were the most dominant threat types. At that time, the main malware distribution technique was infection propagation among computers. The main goal of attacks was to cause damage or to "show off". Since then, we have seen the emergence of new attack types that reflect either new distribution techniques (e.g. web attacks) or that mainly have monetary goal (fake anti-viruses, adware/spyware). 61% of attacks are transmitted from an exploiting machine, 37% of attacks are transmitted from malicious websites, 2% are from other sources not specified.

The results confirmed the hypothesis that most attacks are small, and so low severity threats. This result informed other research in the lablet. It is important to recognize though that Symantec (as an example of key anti-virus vendor) prioritizes

releasing attack signatures for fast propagating threats, but not necessarily for threats that might cause high damage. We conducted a global assessment of the extent to which countries were threatened, threatening, or utilized for sending cyber-attacks. It was hypothesized that once GDP, internet penetration, and the number of attacks are accounted for there are no differences by country. This hypothesis was disconfirmed. While these factors do mitigate the effect, there are still country effects.

Developed countries are more likely to encounter web attacks and fake applications (such as fake anti-viruses) – see Figure 1. High ICT countries, e.g., US, are more likely to encounter fake applications and web attacks; whereas, mid ICT countries are more likely to encounter threats. Mid ICT countries are most likely to transmit attacks, e.g., Romania, Moldova, Bosnia. And in general, the USA is exposed to more fake applications than other countries. It is possible that attackers target these countries because such attacks are likely to be more lucrative. From a statistical perspective – monetary and computing resources are the main factor that attracts attacks at the country level.

Further, we find that the global threat profile is complicated and does not match the standard view from political science of the US and one or two other countries mostly fighting with each other. This means that our results are counter to the standard wisdom which is case based. A general social-influence model is a better predictor than the traditional great-powers model. How attacking computers' hosting varies across countries was analyzed and factors that explain such variation were identified. We found that many countries in Eastern Europe and Central America extensively host attacking computers. Such countries have a combination of good computing infrastructure and high levels of corruption. The high levels of corruption facilitates conducting cyber criminal activities such as registering malicious websites through the complicity of ISPs and law officials.

The international cyber attack network was analyzed. For web attacks and fake applications, most attacks are from Eastern Europe and Central America to developed countries in Western Europe and North America. See for example, the network of dominant attacks from the Ukraine in Figure 2. Exploits have a tendency to spread to geographically nearby countries.

The country-level specificity of the results means that diplomatic and soft-power solutions may be valuable in mitigating cyber-attacks. Many countries are cyber-crime friendly environments – see Figure 3. Some of these countries serve as wayports – and so are “usable” by others to send attacks. Globally – countries with weak cyber policies or poor enforcement or unsophisticated approach to cyber attacks are most “used” or serve as the “source” to spread attacks. Thus, countries in Eastern Europe and Central America host most cyber-attack infrastructure (such as malicious web sites and botnets). A combination of good computing infrastructure and lax policies makes the above countries attractive for hosting attack infrastructure. It is interesting to note that the Ukraine and several countries that were part of the USSR fall into this category. We note that Russia is about to require all blogs to register and to add more control over the web. Based on this research and other research we have done on social media we expect this to a) impact the potential for state instability, and b) to alter the flow of web-based attacks that flow through the related countries. A possible future study might look specifically at the Russia/China/US cyber environment in more detail.

Bibliography

- Broadhurst, Roderic, 2006, "Developments in the global law enforcement of cyber-crime", *Policing: An International Journal of Police Strategies & Management*, 29(3): 408 – 433
- BSA, 2010, Global Cyber-Security Framework. Accessed from: http://www.bsa.org/country/Public%20Policy/~media/Files/Policy/Security/CyberSecure/Cybersecurity_Framework.ashx
- Byres, Eric and Justin Lowe, 2004, The Myths and Facts behind Cyber Security Risks for Industrial Control Systems, Proceedings of the VDE Kongress, 2004.
- Kshetri, Nir, 2005, Pattern of Globalcyber War and Crime: A Conceptual Framework, *Journal of International Management*, 11 (4): 541–562.
- Richardson, R., 2008, 2008 CSI Computer Crime and Security Survey. Accessed from: <http://gocsi.com/sites/default/files/uploads/CSIsurvey2008.pdf>
- Symantec, 2011, 2011 State of Security Survey. Accessed from: http://www.symantec.com/content/en/us/about/media/pdfs/symc_state_of_security_2011.pdf
- Verizon, 2012b, 2012 Data Breach Investigations Report. Accessed from: http://www.verizonbusiness.com/resources/reports/rp_data-breach-investigations-report-2012_en_xg.pdf
- Wynne, Michael W., 2010, Report from the Stevens Institute Cybersecurity Policy Conference, January 19-20 Reagan Building Washington DC.

Kathleen M. Carley - Learned Resiliency in Multi-Level Systems

**See Attachments for graphs

Introduction

Increasingly organizations are under cyber-attack. These attacks may take many forms, with denial of service, data stealing or sabotage, being simple examples. These attacks have the potential to impede mission planning, reduce performance, and cause information to “leak” or “be-discovered” where inappropriate. The question arises, how can organizations be structured to mitigate these cyber-induced risks, to be resilient in the face of these cyber-threats? Using agent-based simulation these questions are addressed. The results provide guidance for how to design for cyber-security resilience. In addition to the

organizational design guidance, this work led to a novel agent-based technology, new metrics of resilience, and new visualization capabilities for simulation data.

Statement of the problem studied

Organization's face security risks from both human and computer (hardware or software) errors. These errors can compound, and grow worse as more individuals are connected to more people and more information technology (IT) in more ways (e.g., VoIP, direct interaction, email, social media ...). Identifying the source of the errors, and responding to errors in a rapid and effective manner is often made more difficult when the organization is operating simultaneously at multiple levels of security – an example is the unclassified, classified secret and classified above-secret levels employed by many governments. The multi-level security impacts IT system, information and human segmentation due to controls on who can relay and access what information using what systems. The need to operate with multi-level security places added constraints on what organizational design solutions are possible to achieve high levels of resiliency in the face of cyber threats and events and increases the need to be resilient in the face of security risks. We ask, is it possible to design organizations that are operating at multiple levels of security to be resilient to cyber threats?

Objective

The objective of this project is to develop a theory of system resiliency for complex adaptive socio-technical systems. A secondary objective is to develop the modeling framework and associated metrics for examining the resiliency of complex socio-technical systems in the face of various cyber- and non-cyber-attacks, such that the methodology can be used to support both basic level simulation based experimentation and assessment of actual socio-technical systems. To meet these objectives multi-modeling is used to examine how to design and impact complex organizational systems in which information is segmented to multiple levels of security with all the attending issues for personnel access and IT usage.

Background

Much of the recent research on threatened complex socio-technical systems, in the security area, has focused on tracking (Lipson, 2002; Gao & Ansari, 2005), investigating (Nilson & Larson, 2008), characterizing (Kotapati et al, 2005), or measuring the damage caused by (Lala & Panda, 2001) cyber attacks. Such work, while providing a context for this research, does not address the issue of system resiliency in the face of such threats. Current approaches seeking to mitigate these threats, in complex socio-technical systems tend to either model cyber attacks at the IT level or provide guidelines to managers about how to handle the human side of the equation. For example, on the IT side, the CAML approach to modeling cyber attacks focuses on data streams and features of the IT system (e.g., Cheung, Lindqvist & Fong, 2003), and the DDoS attack detection models focus on network usage characteristics (Li, Li and Jiang, 2008). For example, on the organizational side, efforts have been made to provide managers with exercises for improving reaction to threat events (Andersen et al., 2004). One exception to this trend is the cyber command and control model (Scherrer & Grund, 2009) which is a conceptual model laying out processes and lines of authority for the DOD and which cannot be used for empirical assessment of resiliency, nor is it generally applicable beyond the DoD. We seek to develop a model that considers both the human and the IT side, such that the model supports empirical analysis of both hypothetical and real complex socio-technical systems. The other key exception is the service restoration model (Lee, Mitchell & Wallace, 2007) which uses highly theoretical and stylized organizations subject to generic attacks and assesses resilience along seven dimensions. We build on these dimensions but recast them, and formalize them, using network metrics.

In the area of cyber threat, relatively less is known about the human than the IT side of the equation (Stytz & Banks, 2008). In general, more work is needed on how humans and so organizations respond to varying cyber threats and events – especially those of significant impact and duration. Drawing on the work in high-reliability organizations, there are models of mitigation based on assumptions of human error and organizational design (LaPorte & Consolini, 1991; Bigley & Roberts, 2001). Admittedly the high reliability research was focused on types of threat other than cyber; nevertheless, this research speaks to the need for specialized organizational structures and norms when the socio-technical system must exhibit high reliability in the face of a highly volatile and potentially hostile environment. Organizations faced with high security needs are in a similar environmental situations and so too would need specialized structures and norms. In complex socio-technical systems, where security is a premium, a comprehensive model must account for both these IT and organizational issues. The proposed work is a step toward creating a joint human-IT framework for assessing the impact of cyber threat and the resiliency of the socio-technical system.

Current approaches to understanding the impact of threats and vulnerabilities on these systems tend to focus on the system as it was planned with some flexibility for how it was built – rarely do cyber or organizational security attempt to identify how the system(s) may evolve. However, the nature of the threat evolves, the threatened organization, its operating environment and its operating dependencies are changing, and the individuals in the organization are learning. While a full explication of these co-evolutionary processes is beyond this proposal we do focus in on the extent to which individual learning results in the evolution of trust, norms, and coordination practices that may support or be detrimental to overall system resiliency. For this we employ agent-based models of learning. The impact of learning on system behavior has been extensively studied. Key finding of relevance here are that: the type of learning employed impacts outcomes (Lant & Mezas, 1992); learning at the individual and group level can conflict resulting in a decrease in resiliency (Carley & Svaboda, 1996); organizational-complexity and the organizational structure impacts the leaning of norms (Harrison & Carroll, 1991) and so would impact the development of a security culture; in contrast to novices, experts models of complex phenomena are richer (Chi, Feltovich & Glaser, 1981),

contain more detailed relations among factors (Klein & Hoffman, 1993) informed by experience (Klein, 1998) and so the experts should be less likely to under or over-react than novices; high personnel turnover in conjunction with a volatile and perhaps hostile environment can mitigate the impact of learning and decrease resilience (Lin & Carley, 2003).

Approach

To address these issues, an approach combining agent-based simulation and dynamic network analysis is used. The specific agent-based model that we extended is Construct (Carley 1990; Carley, Martin & Hirshman, 2009). The result was a new system that supports reasoning about both individuals and groups and can be used to assess multi-level security systems at the organizational level. Resiliency metrics based on a high-dimensional dynamic network representation of organizations were constructed. This representation is referred to as a meta-network representation of the socio-technical organization. A series of scenario driven virtual experiments were then used to assess the relative resiliency of organizations with different designs in either or both the lines of personnel authority and interaction or IT/data access. Over 2 million simulation runs were conducted generating over 900GB of simulated data. A response-surface visualizer was developed for visualizing the results, a set of compression and aggregation techniques were developed for data management. A new system for running the models through Condor was developed.

Summary of Key Results

These simulations show that most organizations are reasonably resilient to small and medium cyber attacks. The attack must be large and fairly pervasive to have a major impact on performance and the ability to engage in and complete mission planning. We find this result to be consistent with the data findings from Symantec. Our results further indicate that hierarchies, overall, are among the top performers and exhibit high resilience when operating a multi-level security environment. When under cyber-attack resiliency is enhanced by organizations resort to direct human communication; however, that increases the chances of inadvertent information leaks. Results of these simulations further suggest that inadvertent information leaks are more likely to occur when the organization is under cyber-attack and are likely to occur in all organizations, regardless of their design. These can be thought of as “normal” accidents. However, such leaks, are most likely in mesh organizations and least likely in hierarchies – see Figure 1.

At one level, these results suggest that organizations to be resilient in the face of cyber-attacks should operate as a high-reliability organization (Weick & Roberts, 1993; Shulman, 2004; Roberts, 1990). Such organizations are ones that utilize management and design practices that enable them to avoid failure despite operating in high risk environments where errors can be expected due to both the complexity of the system and the level of external risk. High-tech multi-level security systems are inherently complex and the potential cost of errors due to cyber threats creates a high risk environment. In these simulations, those organizational designs that rely on personal expertise, and that support change in the face of attacks (commitment to resilience) operate at higher reliability.

At another level, these results refine the notion of what it takes to be a high reliability organization by providing explicit guidance for how to design for resiliency. In particular, our results indicate that:

- Redundancy leads to improved performance and resiliency but at the cost of increased opportunities for information leakage.
- Hierarchies are more impacted initially by a reliability attack, but are barely impacted by an integrity attack. Scale Free organizations are the opposite. In the absence of an attack, the Scale Free organizations have superior task performance.
- Although leadership may be relatively insulated from attacks, specific sub-populations of interest within the organization will be more impacted. IT and Human IT changes tend to improve the ability of the hierarchical organization to support these sub-populations, while such changes in the scale-free organizations are detrimental.
- Low magnitude attacks are unlikely to be noticed by leadership unless leadership is looking for them. To maintain high reliability, leadership needs to be vigilant to these attacks.
- Organizations with IT dependencies are able to shrug off minor attacks because information is not optimally distributed for efficiency.
- Increasing the number of information classification levels and distribution protocols, degrades robustness.
- Combinations of attacks are more harmful than single attacks.
- Cloud topologies suffer more in the short-term, but are more robust in the long.
- Hierarchical organizations with cloud IT are the most robust tested organization in the long-term.
- Stove-piped IT systems, where each system is maintained separately, tend to retain bad information longer and so are less resilient in the face of integrity attacks.
- Matrix Organizations, with their cross-functional teams, may be able to overcome knowledge gaps caused by cyber-attacks; but, are often to coherently finish any tasking once the attack has begun.

At a technical level, a key finding is that agent-based simulation modeling that employs “social” reasoning and so agents at both the individual and group level is a significant win as it enables increased accuracy, increased predictive capability in general, and increased speed/ number of actors modelable. Traditionally in agent-based modeling as you improve the model by making the agents more cognitively accurate, or by making the networks more realistic the number of agents that could be modeled or the speed of the model runs decreased. Our results demonstrate that adding social cognition to the model and the associated multi-level actions actually increased the number of agents modelable or the speed for the same number of agents, in the cyber-security domain.

At a measurement level, this research led to a temporal approach and a set of network of metrics for assessing organizational

resiliency to cyber attacks and other organizational issues – see Figure 2. The basic idea is that a metric of interest can be used to assess immediate impact, persistent impact and recovery by varying the time period of interest. These metrics take the into account the lines of authority and communication among personnel, access to data and IT systems, mode of communication, and direct IT to IT connections. Both static and dynamic metrics were developed. In general, we find that the dynamic metrics are more effective for assessing resiliency than the static.

Bibliography

- Andersen, David, Dawn M. Cappelli, Jose J. Gonzalez, Mohammad Mojtahedzadeh, Andrew P. Moore, Eliot Rich, Jose Maria Sarriegui, Timothy J. Shimeall, Jeffrey M. Stanton, Elise A. Weaver, and Aldo Zagonel. 2004. Preliminary System Dynamics Maps of the Insider Cyber-threat Problem. Paper read at System Dynamics Modeling for Information Security: An Invitational Group Modeling Workshop, 16-20 February, at Pittsburgh, PA, USA.
- Bigley, Gregory A. and Karlene H. Roberts. 2001. The incident command system: High-reliability organization for complex and volatile task environments. *Academy of Management Journal*, 44, 6, 1281-1300.
- Carley & Svoboda, 1996. Kathleen M. Carley & David M. Svoboda, 1996, Modeling Organizational Adaptation as a Simulated Annealing Process. *Sociological Methods and Research*, 25(1): 138-168
- Carley, Kathleen M., 1990, "Group Stability: A Socio-Cognitive Approach," *Advances in Group Processes: Theory and Research*. Edited by Lawler E., Markovsky B., Ridgeway C. and Walker H. (Eds.), Vol. VII. Greenwich, CN: JAI Press, 7: 1-44.
- Carley, Kathleen M., Michael K. Martin and Brian Hirshman, 2009, "The Etiology of Social Change," *Topics in Cognitive Science*, 1.4:621-650.
- Zhiang Lin and Kathleen M. Carley, 2003, *Designing Stress Resistant Organizations: Computational Theorizing and Crisis Applications*, Boston, MA: Kluwer.
- Cheung, S.; Lindqvist, U.; Fong, M.W.; , "Modeling multistep cyber attacks for scenario recognition," *DARPA Information Survivability Conference and Exposition*, 2003. *Proceedings* , vol.1, no., pp. 284- 292 vol.1, 22-24 April 2003
- Chi, M.T.H., Feltovich, P.J., & Glaser, R. , 1981. Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5, 121-152.
- Dennis K. Nilsson and Ulf E. Larson. 2008. Conducting forensic investigations of cyber attacks on automobile in-vehicle networks. In *Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, and multimedia and workshop (e-Forensics '08)*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium
- Gao, Zhiqiang and Ansari, N., 2005 , "Tracing cyber attacks from the practical perspective," *Communications Magazine, IEEE* , vol.43, no.5, pp. 123- 131, May 2005
- Harrison, J.R. and G.R. Carroll. 1991. Keeping the Faith: A Model of Cultural Transmission in Formal Organizations. *Administrative Science Quarterly*, 36, 552-582.
- Kameswari Kotapati, Peng Liu, Yan Sun and Thomas F. LaPorta, 2005, A Taxonomy of Cyber Attacks on 3G Networks *Intelligence and Security Informatics Lecture Notes in Computer Science*, Volume 3495/2005, 129-138
- Klein, G. A., & Hoffman, R. R., 1993. Seeing the invisible: Perceptual-cognitive aspects of expertise. In Rabinowitz, M. (ed.), *Cognitive science foundations of instruction*. Hillsdale, NJ: Erlbaum. 203-226.
- Klein, Gary A. 1998, "Sources of Power: How People Make Decisions", MIT Press, Cambridge, Mass, pp. 1-30.
- La Porte, Todd R. and Paula M. Consolini. 1991. Working in Practice But Not in Theory: Theoretical Challengers of 'High-Reliability Organizations'. *Journal of Public Administrative Research and Theory*, 1, 1, 19-47.
- Lala, C. and B. Panda, 2001, Evaluating damage from cyber attacks: a model and analysis, *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 31(4):300-310.
- Lant, T.L. and S.J. Mezias, 1992, "An Organizational Learning Model of Convergence and Reorientation," *Organization Science*, 3(1): 47-71.
- Lee, Earl E. II, John E. Mitchell, and William A. Wallace. 2007. Restoration of Services in Interdependent Infrastructure Systems: A Network Flows Approach. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 37 (6):1303-1317.
- Li, Muhai, Ming Li and Xiuying Jiang, 2008, DDoS attacks detection model and its application, *WSEAS Transactions on Computers*, 7(8).
- Lipson, Howard F., 2002, *Tracking and Tracing Cyber-Attacks: Technical Challenges and Global Policy Issues*, Technical Report, Software Engineering Institute, Carnegie Mellon University
- Roberts, K. H. (1990). Some Characteristics of High-Reliability Organizations. *Organization Science*, 1, 160-177.
- Scherrer, Joseph H., and William C. Grund. 2009. *A Cyberspace Command and Control Model*. edited by U. A. Force. Maxwell AFB, AL: Air War College.
- Schulman, P. R. (2004). General attributes of safe organizations. *Quality and Safety in Health Care*. 13, Supplement II, ii39-ii44.
- Stytz, Martin & Sheila Banks, 2008, *Advancing Cyber Warfare Simulation System Capabilities*, SimTecT 2008 Simulation Conference: Simulation - Maximising Organisational Benefits (SimTecT 2008) Melbourne, Australia, May 12 – 15 , 2008
- Weick, K. E., & Roberts, K. H. (1993). *Collective Mind in Organizations: Heedful Interrelating on Flight Decks*. *Administrative Science Quarterly*, 38, 357-381.

****See Attachment**

Anupam Datta, Limin Jia - Secure Composition of Systems and Policies

Problem studied:

Our research aims to develop compositional reasoning principles to verify systems that consist of trusted and adversarial components. In particular, our adversaries can supply code to be executed by trusted components.

Results:

We developed a program logic that can reason about systems that contain trusted and untrusted components. In particular, untrusted components can provide code to be executed by trusted components: e.g., via modifying the code region of trusted programs. Our program logic contains two novel typing rules that derive security properties of two commonly-used security mechanisms for executing untrusted code: sandboxing and code identification. We proved our program logic sound with regard to a step-indexed semantics. We demonstrate the expressiveness of our program logic by verifying the security property of the design of Memoir [1], a previously proposed trusted computing system for ensuring state continuity of isolated security-sensitive applications.

Robert Harper - An Epistemic Formulation of Information Flow Analysis

Problem studied:

We continue our investigation of using epistemic logic to formalize information flow analysis. We also began exploring applying epistemic logic to analyzing information flow in social networks.

Results:

The main development in the research is that we have begun to consider timing channels as these are important when there is a centralized certificate authority that interacts with multiple security sensitive processes. Timing channels are difficult to avoid as any branching on confidential inputs can create a timing leak if the branches do not take the exact same time to execute. We are seeking a reasonable approach that minimizes the probability of timing leaks by adding sleep delays into the program where necessary.

In addition, as a primarily exploration, we have extended the linear epistemic logic in [2] to analyze information flows on social networking sites such as Facebook. More concretely, we extended [1] with names, a dedicated mechanism for global information, list comprehension and membership testing for lists to model common constructs in social networking sites, such as access control lists.

Frank Pfenning - Proofs and Signatures

In the current year, Frank Pfenning, Dennis Griffith and Elsa Gunter have made significant strides both in the theory and implementation of a session-typed language (called SILL) for secure, distributed programming. On the theoretical side, we have integrated various form of resource management into the formal description of the language and established its fundamental properties. We have also developed techniques for reasoning about programs in the language using parametricity, which is particularly important in this setting since many security properties of programs are consequences of parametricity. On the implementation side, we have constructed and continually refined the prototype. This includes a much improved front-end, integrating new techniques for combining the type system of an ambient host language and the concurrency module. It also includes several back ends that are appropriate in different circumstances: two of them employs shared memory for synchronous or asynchronous communication, a third one is distributed and uses explicit message passing. This prototype has been used successfully in teaching at a summer school affiliated with a multi-site European project on behavioral types. We have also designed techniques to dynamically check the adherence of processes to prescribed communication protocols under a practical adversary model. This latter design is critical for purposes of technology transfer, but has not yet been integrated into the implementation.

Main Highlights

We have designed a programming language for secure distributed computation which supports both formal proof and digital signatures in order to establish trust.

We have established crucial properties of this language, such as session fidelity, deadlock freedom, type preservation, parametricity, and termination.

We have implemented a prototype that allows us to write, check, and execute concurrent, and distributed session-typed programs.

We have designed and implemented a new type inference algorithm that modularly combines a background language with its concurrent extension.

We have designed and implemented an automatic resource control regime that is fully logically justified.

We have developed a system of dynamic checking and blame assignment towards a theory of causality and accountability in distributed computation.

André Platzer - Security Reasoning for Distributed Systems with Uncertainty

INTRODUCTION

This project is focused on understanding large systems in the presence of adversarial attacks, stochastic component failures and benign stochastic noise. The intuitive challenge is that sensor or measurement noise is ubiquitous, and noise makes it difficult to determine whether an anomalous component has been maliciously manipulated or has permanently failed. However, a reliable system should be robust to disruptions—we should design systems to shrug off both failure and attack, and still efficiently achieve their intended goal.

This is a critical security issue that will become increasingly important as autonomous and semi-autonomous devices—including cyber-physical systems (CPSs) like unmanned aerial vehicles (UAVs) and other robotic systems—become more widely adopted in governments, economic processes and militaries. Security vulnerabilities in these systems will cause serious real-world effect. Worryingly, security vulnerabilities in this setting could be extremely subtle.

As a speculative example: suppose that a particular UAV became a popular delivery vector for small packages in the near future. If a hostile entity were able to somehow manipulate these devices—say by remotely tampering with sensor calibration—then this might be a multi-million dollar attack in terms of lost productivity. This manipulation might be something as explicit as inducing the UAVs to crash into the ground or other UAVs, but might be less obvious such as reducing the UAV's delivery efficiency. We want to develop techniques that can assess how susceptible a CPS is to manipulation. Furthermore, we want to invent techniques that allow use to automatically design controllers and policies for autonomous (or semi-autonomous) systems that are efficient, reliable and difficult to manipulate.

In previous years we have already made progress on aspects of this topic. In last year's ARO annual report we discussed # SAT. # SAT is a new computational problem designed to help analyze robustness in the presence of failure. A large class of reliabilities and robustness questions can be expressed as # SAT instances.

For example, we can model many deterministic planning problems in Boolean propositional logic. Once this has been done, we can declare parts of the model to be 'prone to failure' with a certain failure probability. Our interest is determining how likely is it that we can achieve an objective given independent failure of these failure-prone components. If the percentage is too low, this suggests that the system is not robust to the type of failures modeled, and should be redesigned with more redundancy.

In our paper "A generalization of SAT and #SAT for robust policy evaluation" we provided both a theoretical analysis and empirically successful algorithm for solving # SAT problems exactly.

This year we have been exploring the issue of controller synthesis. Our algorithm for # SAT is designed to estimate how robust a moderately-large design is to stochastic component failure. By itself it can be used as an inner loop of a heuristic controller design algorithm—some kind of local search, for example. Our work this year has been aimed at explicitly designing reliable controllers for systems that are much too large to handle using exact # SAT methods while, at the same time, providing explicit bounds on the quality of the policy.

SUMMARY OF MOST IMPORTANT RESULTS

We had several major results this year. They are summarized in our NIPS workshop paper "A projection algorithm for strictly monotone linear complementarity problems", and partially expounded upon in our journal paper "Hybrid theorem proving of aerospace systems: applications and challenges". We expect to complete a new paper on this topic relatively soon, as well.

Our principle focus this year has been on approximating large and continuous decision problems. We are particularly interested in control and decision problems stemming from CPS security. Robotic systems are good examples of a CPS. As a running example of a CPS we will use the problem of designing a controller for a UAV. However, part of the promise of our work is that it is a fundamental exploration into planning under uncertainty, adversarial disruption and robustness—the research will apply to many other security problems that bare little superficial resemblance to our UAV example.

Designing a controller for a UAV that maintains good performance in the face of benign uncertainty—e.g. uncertainty stemming from sensor noise and actuator imprecision—is an example of a core problem that has been explored from a number of different fields. Here the problem is based on finding a mapping from a UAV's current and past sensor measurements into an

appropriate set of signals to its actuators that achieves some goal. For example, we want to determine a way of building a controller that takes in noisy sensor information (that could include GPS information, a camera feed, and information from an inertial measurement unit) and determine how this can be used to guide the UAV from point A to point B quickly without crashing into any other object along the way.

There are several existing ways of approaching this controller design problem. A standard way is to model the continuous flight dynamics of the UAV (using, for example, differential equations), then to discretize and linearly approximate these dynamics. This discrete linearization can be turned into a Markov decision process (MDP)—a popular formalism for sequential decision making. Once the MDP has been formulated, an optimal policy for controlling the UAV can be found using dynamic programming algorithms.

One thing to note about this process is that by discretizing the system we have approximated the true continuous dynamics of the system. How accurate is this approximation? Any approximate representation degrades solution quality. The difference between the solution to the approximate problem and the solution to the ideal problem is called representation error.

There is a natural tradeoff, in the above example, between how finely state-space is gridded—and hence how accurate the approximation is—and how difficult it will be to find a policy with dynamic programming. As the number of continuous dimensions increases, the number of states in a finely-gridded representation will explode. Bellman dubbed this the curse of dimensionality in the 1950s. Finding a good control policy in continuous problems with dimensions much higher than 8 is a serious problem even for modern computers.

The above discussion raises some natural questions. How sensitive is the optimal discrete policy to the discretization schemes—if we grid more coarsely, how much worse will the optimal discrete policy be when applied to the continuous setting? Are there other ways to approximately represent the problem other than gridding? When do they offer a better trade-off between representation size and representation error? Our work this year has been focused on characterizing these representational issues in a broad class of decision problems and a general form of approximate representation.

We have been working on monotone linear complementarity problems (LCPs). Any decision problem with linear or convex quadratic objectives and linear constraints can be represented as a monotone LCP. MDPs are LCPs, so the above discussion on UAVs can be discussed using LCPs. Furthermore, linear programs (LPs; a nearly ubiquitous class of optimization problems), support vector machines (SVM; a classification model used widely in machine learning applications), and any convex quadratic program (QPs; a more general optimization framework than LPs) can all be cast as monotone LCPs.

A LCP, formally, is described by a square matrix A and a vector b . Monotonicity is a technical condition on A that is important for ensuring the LCP is tractable. A vector x is a solution to the LCP if three conditions hold, which can be concisely written as: $x \geq 0$, $Ax - b \geq 0$, and $x^T(Ax - b) = 0$. In other words, x must be non-negative, its affine transformation $Ax - b$ must be non-negative, and x must be orthogonal to $Ax - b$.

Together, the non-negativity and orthogonality tell us that for a solution x , either $x_i = 0$ or $(Ax - b)_i = 0$.

Readers familiar with optimization might notice a similarity between the definition of a solution to an LCP and KKT conditions—which are necessary and (usually) sufficient first-order conditions for optimality in convex programs. This is no accident: the connection between optimization and complementarity problems is deep. We will omit detailing this connection here, however. We note that the definition of an LCP can be extended to the continuous setting where A becomes a linear operator and both x and $Ax - b$ are functions in some function space, but we will neglect to do so in this document. This may be important for dealing with continuous decision problems, however, and we are actively working in this setting.

For a large or continuous LCP we need to approximate A and b . We do this with a general technique that can be called using a finite basis expansions. This representation is simple to describe, we represent a function as a linear combination of basis functions: $f(x) = \sum_{i=1}^n c_i \phi_i(x)$. Here, c is the vector that we are trying to find, ϕ is the vector of fitting coefficients and ϕ is a set of basis vectors. When n is small, f is a low-dimensional representation of f —we are representing the numbers of c with the numbers in ϕ . In order for this low-dimensional representation to be relatively accurate we want ϕ to be the projection of f onto the range of ϕ , denoted by $\text{range}(\phi)$, which is the closest approximation of f possible using the basis ϕ . This ideal of projection onto a low-rank space is central to our idea of approximation.

Our major contribution has been develop an approximate iterative algorithm for solving strictly monotone LCPs that uses these low-dimensional representations. This algorithm detailed in our paper “A projection algorithm for strictly monotone linear complementarity problems.” This algorithm is much faster than previous iterative algorithms for the same problem, and we have proved bounds on the approximation error that it induces—it is very accurate as long the actual solution of the LCP x^* is not far from its projection $\Pi_{\text{range}(\phi)}(x^*)$.

Critically, we have a stochastic version of the algorithm no longer has dependency on the size of the problem n . This means that our algorithm can be totally liberated from issues associated with discretizing a continuous problem. It is based on sampling while it is running rather than using a ‘mesh’ of the state-space.

We consider this to be an important advance for any security problem that involves planning under uncertainty, such as any security problem involving robots, because it is a fast approximate, but also maintains an explicit bound on its approximation error. Exact solvers will be too slow to solve many problem instances, and many other approximate solvers do not have approximation guarantees—it will not be possible to use their solutions to formally reason about the security properties.

FUTURE DIRECTIONS

In the coming year we will primarily work on implementation of our algorithms, and extending them to more general settings. Our algorithms currently work on strictly monotone LCP, and we want to extend them so that they also apply to non-strictly monotone LCPs. Monotone LCPs can model more problems than strictly monotone LCPs, although there is usually a strictly monotone LCP approximation of any monotone LCP that can be formed with a simple regularization process.

We also want to showcase how our algorithm performs on a continuous two UAV collision problem. The goal is to find a good collision-avoiding control policy for a UAV using different high-dimension model of flight dynamics. We expect that our approach will be significantly faster than existing dynamic programming approaches and, again, will have the virtual of carrying formal bounds on its quality.

Jurgen Pfeffer - Composability of Big Data and Algorithms for Social Networks Analysis Metrics

**See Attachments

Technology Transfer

Kathleen Carley - Learned Resiliency in Multi-Level Systems

1) CASOS Summer Institute, June 2013 - This is technology transfer through education. We trained over 45 individuals in a) the new metrics, b) the process of assessing organizational-level impacts of cyber events, and c) use of the multi-level simulation technology developed for this project. Trainees included individuals from universities, government and industries. Due to sequestration there were no active duty military participants this year.

2) STRATCOM - We engaged in discussions with members of STRATCOM about this research with the aim of providing them with material needed for their design review process.

3) USMA - Michael Lanham has returned to USMA and will be teaching Cadets using the materials and models developed herein. He is part of the cyber group.

Kathleen Carley - Geo-Temporal Characterizations

1) CASOS Summer Institute, June 2013 - This is technology transfer through education. We trained over 45 individuals in a) procedure for assessing global cyber network, b) over time analytics and MRQAP needed to assess global cyber network, and c) result of global cyber network assessment.

2) STRATCOM - We engaged in discussions with members of STRATCOM about this research with the aim of providing them with material requested on global cyber networks.

3) Discussions with NATO about possible international issues related to cyber security. We expect to be invited to a conference in Belgium on this issue in October 2013.

4) Other NAS projects – results from this study were leveraged to validate model developed under the Learned Resiliency project.

David Garlan, Jonathan Aldrich, Bradley Schmerl - Secure Frameworks

1) We have successfully transferred the Rainbow framework to our partners at GMU.

2) An open-source prototype of the system described in the GlobalDSL paper is available at <https://github.com/wyvernlang>

Geo-Temp Characterization - 29983.10

Kathleen M. Carley

Introduction

Objective

The objective of this project is to empirically characterize the nature of the current threat environment and to test a series of existing hypotheses about that threat environment using Symantec data. Our focus is global. The basic theory is that the potential severity of the threat is a function of the political environment rather than the technology. Questions to be addressed empirically include:

1. What is the likelihood of a catastrophic threat? Hypothesis: Most attacks are small.
2. How does the likelihood differ by type of threat? Hypothesis: there are no differences by type of threat.
3. Do these answers differ by country? Hypothesis A: Once GDP, internet penetration, and the number of attacks are accounted for there are no differences by country.
Hypothesis B: The likelihood of a company being attacked depends on their position in the alliance/enmity network.

Background

In their 2011 survey Symantec found that the number one cyber risk business concern was external cyber-attacks, followed by concerns about both unintentional insider error (2nd risk) and intentional insider error (3rd risk) (Symantec, 2011, pg 9). Analysis by Verizon's cyber forensics team indicates that the massive increase in external threats overshadows insider attacks (Verizon, 2012b, pg 21). See also Richardson (2008). Despite the increase in external threats little is known about the source of such threats; or the global implications this evolving threat environment.

Wynne (2010) notes that the need to do attribution and forensics is critical to stem the tide of cyber-attacks. To meet this need, an understanding of the threat environment at the global level is needed. Cyber security, at the global level, is critical on a number of fronts including countering terrorism (Westby, 2007). At the global level, cyber security requires not only attribution and forensics, but harmonized laws and effective information sharing. In spite of this growing consensus there is still little empirical understanding of the global cyber threat environment, an understanding that is critical for forensics. We have found that most global information sharing is done through security providers and the data is only now becoming available and in restricted form given the dual privacy and security needs that must be met.

We note that reliance on anecdotal evidence can be damaging for the Science of Security. As empirical findings come to light, assumptions about the nature of cyber threat are changing. For example, in 2004, Byre and Lowe showed that process control and SCADA systems were not immune to attack using incidence data. Further basic assumptions are falling as new empirical evidence comes to light. This is creating a new baseline against which forensics can operate. While there are an increasing body of findings focused on specific threats and empirical

assessments of key incidents; there is less understanding of the human socio-behavioral factors, particularly at the global level.

At the global level, multiple conceptual frameworks abound. For example, Kshetri (2005) argues that country level differences in the regulative, normative and cognitive legitimacy of different types of web attacks lead to differences in the extent to which organized crime can use the internet in those countries. BSA (2010) provides guiding principles for global cyber security. Broadhurst (2006) argues that cyber-attacks are traditional crimes in a new venue which makes traditional forensic methods obsolete. And so on. Despite the recognition of the global nature of the cyber threat there is little characterization of that threat.

Approach

The basic approach used was to develop code for extracting country level indicators of attacks and attack paths from the Symantec data. The analysis is based on the Symantec WINE telemetry data set. WINE is a platform through which external researchers can access data sets used within Symantec Research labs. To the best of our knowledge, Symantec is currently the only security company that makes such platform available to external researchers. The telemetry data set consists of attack reports from more than 10 million Symantec customer computers worldwide.

This was then combined with other open-source data to create a global threat profile. An example of such additional open source data is the ICT index circa 2010 for all countries. The ICT index is a combined measure based on 11 indicators including adult literacy, internet access and so on. The data was fused to create a global indicators data set and was then assessed using network analytics and standard statistical procedures. Based on this data a network based model of the impact of hostilities and other factors on geo-cyber-attack-network was developed.

Summary of Key Results

There are two key aspects to this research. The first is the characterization of the threat profile. The second is the assessment of the over-all global threat with respect to cyber attacks.

In terms of the general threat profile, our results led to an empirical characterization of the change in cyber-threat over the past several years. This work indicated that web attacks account for the vast majority of attacks in the IPS catalog. Around 2003, worms and viruses were the most dominant threat types. At that time, the main malware distribution technique was infection propagation among computers. The main goal of attacks was to cause damage or to "show off". Since then, we have seen the emergence of new attack types that reflect either new distribution techniques (e.g. web attacks) or that mainly have monetary goal (fake anti-viruses, adware/spyware). 61% of attacks are transmitted from an exploiting machine, 37% of attacks are transmitted from malicious websites, 2% are from other sources not specified.

The results confirmed the hypothesis that most attacks are small, and so low severity threats. This result informed other research in the lablet. It is important to recognize though that Symantec (as an example of key anti-virus vendor) prioritizes releasing attack signatures for fast propagating threats, but not necessarily for threats that might cause high damage.

We conducted a global assessment of the extent to which countries were threatened, threatening, or utilized for sending cyber-attacks. It was hypothesized that once GDP, internet penetration, and the number of attacks are accounted for there are no differences by country. This hypothesis was disconfirmed. While these factors do mitigate the effect, there are still country effects.

Developed countries are more likely to encounter web attacks and fake applications (such as fake anti-viruses) – see Figure 1. High ICT countries, e.g., US, are more likely to encounter fake applications and web attacks; whereas, mid ICT countries are more likely to encounter threats. Mid ICT countries are most likely to transmit attacks, e.g., Romania, Moldova, Bosnia. And in general, the USA is exposed to more fake applications than other countries. It is possible that attackers target these countries because such attacks are likely to be more lucrative. From a statistical perspective - monetary and computing resources are the main factor that attracts attacks at the country level.

Further, we find that the global threat profile is complicated and does not match the standard view from political science of the US and one or two other countries mostly fighting with each other. This means that our results are counter to the standard wisdom which is case based. A general social-influence model is a better predictor than the traditional great-powers model.

How attacking computers' hosting varies across countries was analyzed and factors that explain such variation were identified. We found that many countries in Eastern Europe and Central America extensively host attacking computers. Such countries have a combination of good computing infrastructure and high levels of corruption. The high levels of corruption facilitates conducting cyber criminal activities such as registering malicious websites through the complicity of ISPs and law officials.

The international cyber attack network was analyzed. For web attacks and fake applications, most attacks are from Eastern Europe and Central America to developed countries in Western Europe and North America. See for example, the network of dominant attacks from the Ukraine in Figure 2. Exploits have a tendency to spread to geographically nearby countries.

The country-level specificity of the results means that diplomatic and soft-power solutions may be valuable in mitigating cyber-attacks. Many countries are cyber-crime friendly environments – see Figure 3. Some of these countries serve as wayports – and so are “usable” by others to send attacks. Globally – countries with weak cyber policies or poor enforcement or unsophisticated approach to cyber attacks are most “used” or serve as the “source” to spread attacks. Thus, countries in Eastern Europe and Central America host most cyber-attack infrastructure (such as malicious web sites and botnets). A combination of good computing infrastructure and lax policies makes the above countries attractive for hosting attack infrastructure. It is interesting to note that the Ukraine and several countries that were part of the USSR fall into this category. We note that Russia is about to require all blogs to register and to add more control over the web. Based on this research and other research we have done on social media we expect this to a) impact the potential for state instability, and b) to alter the flow of web-based attacks that flow through the related countries. A possible future study might look specifically at the Russia/China/US cyber environment in more detail.

Bibliography

- Broadhurst, Roderic, 2006, "Developments in the global law enforcement of cyber-crime", *Policing: An International Journal of Police Strategies & Management*, 29(3): 408 – 433
- BSA, 2010, Global Cyber-Security Framework. Accessed from:
http://www.bsa.org/country/Public%20Policy/~media/Files/Policy/Security/CyberSecurity/Cybersecurity_Framework.ashx
- Byres, Eric and Justin Lowe, 2004, The Myths and Facts behind Cyber Security Risks for Industrial Control Systems, Proceedings of the VDE Kongress, 2004.
- Kshetri, Nir, 2005, Pattern of Globalcyber War and Crime: A Conceptual Framework, *Journal of International Management*, 11(4): 541–562.
- Richardson, R., 2008, 2008 CSI Computer Crime and Security Survey. Accessed from:
<http://gocsi.com/sites/default/files/uploads/CSIsurvey2008.pdf>
- Symantec, 2011, 2011 State of Security Survey. Accessed from:
http://www.symantec.com/content/en/us/about/media/pdfs/symc_state_of_security_2011.pdf
- Verizon, 2012b, 2012 Data Breach Investigations Report. Accessed from:
http://www.verizonbusiness.com/resources/reports/rp_data-breach-investigations-report-2012_en_xg.pdf
- Wynne, Michael W., 2010, Report from the Stevens Institute Cybersecurity Policy Conference, January 19-20 Reagan Building Washington DC.

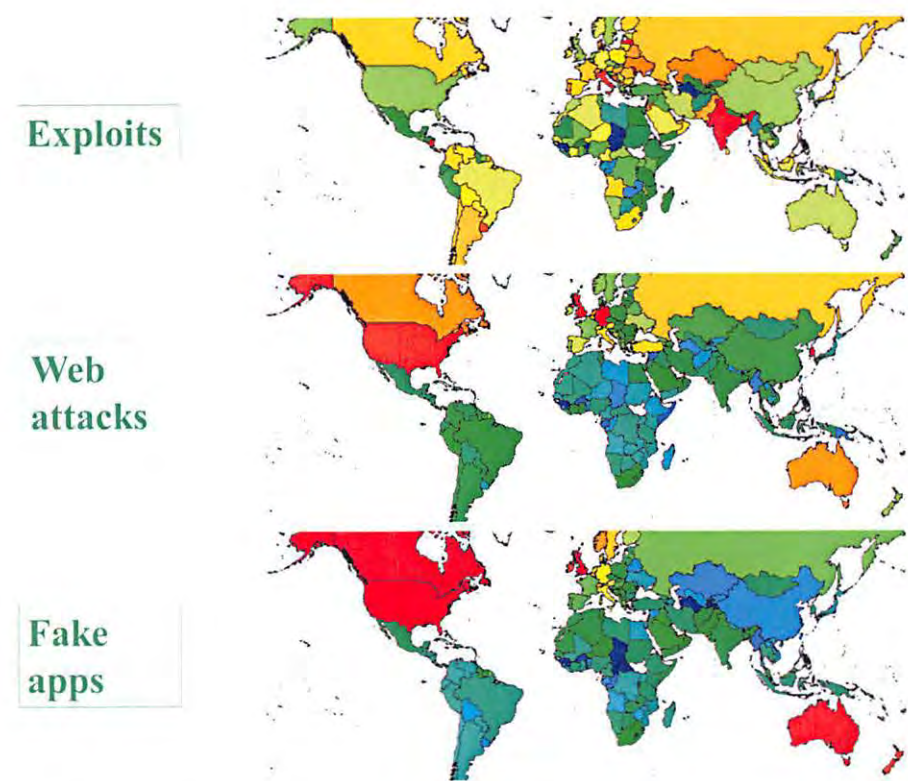


Figure 1. Threatened countries. Red is high, blue is low

ips_ntw



powered by ORA-NetScenes

Figure 3. Those countries receiving many web-attacks from the Ukraine. These countries receive almost 2 standard deviations more attacks from the Ukraine, than the the average number of attacks from one country to another.

Exploits

Web
attacks

Fake
apps

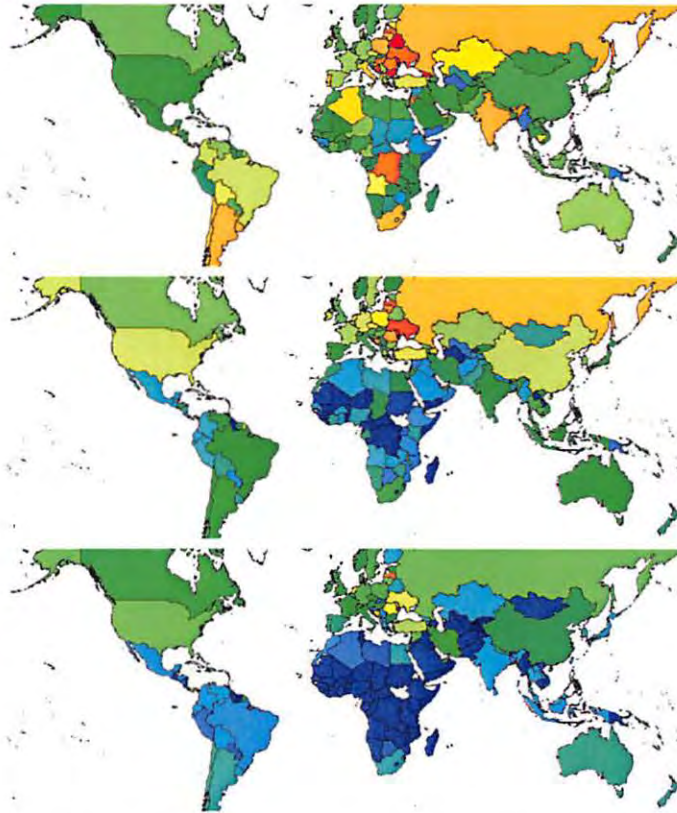


Figure 2. Countries with cyber-crime friendly environment. Red is high, blue is low.

Learned Resiliency in Multi-Level Systems – 26803.11

Kathleen M. Carley

Introduction

Increasingly organizations are under cyber-attack. These attacks may take many forms, with denial of service, data stealing or sabotage, being simple examples. These attacks have the potential to impede mission planning, reduce performance, and cause information to “leak” or “be-discovered” where inappropriate. The question arises, how can organizations be structured to mitigate these cyber-induced risks, to be resilient in the face of these cyber-threats? Using agent-based simulation these questions are addressed. The results provide guidance for how to design for cyber-security resilience. In addition to the organizational design guidance, this work led to a novel agent-based technology, new metrics of resilience, and new visualization capabilities for simulation data.

Statement of the problem studied

Organization’s face security risks from both human and computer (hardware or software) errors. These errors can compound, and grow worse as more individuals are connected to more people and more information technology (IT) in more ways (e.g., VoIP, direct interaction, email, social media ...). Identifying the source of the errors, and responding to errors in a rapid and effective manner is often made more difficult when the organization is operating simultaneously at multiple levels of security – an example is the unclassified, classified secret and classified above-secret levels employed by many governments. The multi-level security impacts IT system, information and human segmentation due to controls on who can relay and access what information using what systems. The need to operate with multi-level security places added constraints on what organizational design solutions are possible to achieve high levels of resiliency in the face of cyber threats and events and increases the need to be resilient in the face of security risks. We ask, is it possible to design organizations that are operating at multiple levels of security to be resilient to cyber threats?

Objective

The objective of this project is to develop a theory of system resiliency for complex adaptive socio-technical systems. A secondary objective is to develop the modeling framework and associated metrics for examining the resiliency of complex socio-technical systems in the face of various cyber- and non-cyber-attacks, such that the methodology can be used to support both basic level simulation based experimentation and assessment of actual socio-technical systems. To meet these objectives multi-modeling is used to examine how to design and impact complex organizational systems in which information is segmented to multiple levels of security with all the attending issues for personnel access and IT usage.

Background

Much of the recent research on threatened complex socio-technical systems, in the security area, has focused on tracking (Lipson, 2002; Gao & Ansari, 2005), investigating (Nilson & Larson, 2008), characterizing (Kotapati et al, 2005), or measuring the damage caused by (Lala & Panda,

2001) cyber attacks. Such work, while providing a context for this research, does not address the issue of system resiliency in the face of such threats. Current approaches seeking to mitigate these threats, in complex socio-technical systems tend to either model cyber attacks at the IT level or provide guidelines to managers about how to handle the human side of the equation. For example, on the IT side, the CAML approach to modeling cyber attacks focuses on data streams and features of the IT system (e.g., Cheung, Lindqvist & Fong, 2003), and the DDoS attack detection models focus on network usage characteristics (Li, Li and Jiang, 2008). For example, on the organizational side, efforts have been made to provide managers with exercises for improving reaction to threat events (Andersen et al., 2004). One exception to this trend is the cyber command and control model (Scherrer & Grund, 2009) which is a conceptual model laying out processes and lines of authority for the DOD and which cannot be used for empirical assessment of resiliency, nor is it generally applicable beyond the DoD. We seek to develop a model that considers both the human and the IT side, such that the model supports empirical analysis of both hypothetical and real complex socio-technical systems. The other key exception is the service restoration model (Lee, Mitchell & Wallace, 2007) which uses highly theoretical and stylized organizations subject to generic attacks and assesses resilience along seven dimensions. We build on these dimensions but recast them, and formalize them, using network metrics.

In the area of cyber threat, relatively less is known about the human than the IT side of the equation (Stytz & Banks, 2008). In general, more work is needed on how humans and so organizations respond to varying cyber threats and events – especially those of significant impact and duration. Drawing on the work in high-reliability organizations, there are models of mitigation based on assumptions of human error and organizational design (LaPorte & Consolini, 1991; Bigley & Roberts, 2001). Admittedly the high reliability research was focused on types of threat other than cyber; nevertheless, this research speaks to the need for specialized organizational structures and norms when the socio-technical system must exhibit high reliability in the face of a highly volatile and potentially hostile environment. Organizations faced with high security needs are in a similar environmental situations and so too would need specialized structures and norms. In complex socio-technical systems, where security is a premium, a comprehensive model must account for both these IT and organizational issues. The proposed work is a step toward creating a joint human-IT framework for assessing the impact of cyber threat and the resiliency of the socio-technical system.

Current approaches to understanding the impact of threats and vulnerabilities on these systems tend to focus on the system as it was planned with some flexibility for how it was built – rarely do cyber or organizational security attempt to identify how the system(s) may evolve. However, the nature of the threat evolves, the threatened organization, its operating environment and its operating dependencies are changing, and the individuals in the organization are learning. While a full explication of these co-evolutionary processes is beyond this proposal we do focus in on the extent to which individual learning results in the evolution of trust, norms, and coordination practices that may support or be detrimental to overall system resiliency. For this we employ agent-based models of learning. The impact of learning on system behavior has been extensively studied. Key finding of relevance here are that: the type of learning employed impacts outcomes (Lant & Mezias, 1992); learning at the individual

and group level can conflict resulting in a decrease in resiliency (Carley & Svaboda, 1996); organizational-complexity and the organizational structure impacts the learning of norms (Harrison & Carroll, 1991) and so would impact the development of a security culture; in contrast to novices, experts models of complex phenomena are richer (Chi, Feltovich & Glaser, 1981), contain more detailed relations among factors (Klein & Hoffman, 1993) informed by experience (Klein, 1998) and so the experts should be less likely to under or over-react than novices; high personnel turnover in conjunction with a volatile and perhaps hostile environment can mitigate the impact of learning and decrease resilience (Lin & Carley, 2003).

Approach

To address these issues, an approach combining agent-based simulation and dynamic network analysis is used. The specific agent-based model that we extended is Construct (Carley 1990; Carley, Martin & Hirshman, 2009). The result was a new system that supports reasoning about both individuals and groups and can be used to assess multi-level security systems at the organizational level. Resiliency metrics based on a high-dimensional dynamic network representation of organizations were constructed. This representation is referred to as a meta-network representation of the socio-technical organization. A series of scenario driven virtual experiments were then used to assess the relative resiliency of organizations with different designs in either or both the lines of personnel authority and interaction or IT/data access. Over 2 million simulation runs were conducted generating over 900GB of simulated data. A response-surface visualizer was developed for visualizing the results, a set of compression and aggregation techniques were developed for data management. A new system for running the models through condor was developed.

Summary of Key Results

These simulations show that most organizations are reasonably resilient to small and medium cyber attacks. The attack must be large and fairly pervasive to have a major impact on performance and the ability to engage in and complete mission planning. We find this result to be consistent with the data findings from Symantec. Our results further indicate that hierarchies, overall, are among the top performers and exhibit high resilience when operating a multi-level security environment. When under cyber-attack resiliency is enhanced by organizations resort to direct human communication; however, that increases the chances of inadvertent information leaks. Results of these simulations further suggest that inadvertent information leaks are more likely to occur when the organization is under cyber-attack and are likely to occur in all organizations, regardless of their design. These can be thought of as “normal” accidents. However, such leaks, are most likely in mesh organizations and least likely in hierarchies – see Figure 1.

At one level, these results suggest that organizations to be resilient in the face of cyber-attacks should operate as a high-reliability organization (Weick & Roberts, 1993; Shulman, 2004; Roberts, 1990). Such organizations are ones that utilize management and design practices that enable them to avoid failure despite operating in high risk environments where errors can be expected due to both the complexity of the system and the level of external risk. High-tech multi-level security systems are inherently complex and the potential cost of errors

due to cyber threats creates a high risk environment. In these simulations, those organizational designs that rely on personal expertise, and that support change in the face of attacks (commitment to resilience) operate at higher reliability.

At another level, these results refine the notion of what it takes to be a high reliability organization by providing explicit guidance for how to design for resiliency. In particular, our results indicate that:

- Redundancy leads to improved performance and resiliency but at the cost of increased opportunities for information leakage.
- Hierarchies are more impacted initially by a reliability attack, but are barely impacted by an integrity attack. Scale Free organizations are the opposite. In the absence of an attack, the Scale Free organizations have superior task performance.
- Although leadership may be relatively insulated from attacks, specific sub-populations of interest within the organization will be more impacted. IT and Human IT changes tend to improve the ability of the hierarchical organization to support these sub-populations, while such changes in the scale-free organizations are detrimental.
- Low magnitude attacks are unlikely to be noticed by leadership unless leadership is looking for them. To maintain high reliability, leadership needs to be vigilant to these attacks.
- Organizations with IT dependencies are able to shrug off minor attacks because information is not optimally distributed for efficiency.
- Increasing the number of information classification levels and distribution protocols, degrades robustness.
- Combinations of attacks are more harmful than single attacks.
- Cloud topologies suffer more in the short-term, but are more robust in the long.
- Hierarchical organizations with cloud IT are the most robust tested organization in the long-term.
- Stove-piped IT systems, where each system is maintained separately, tend to retain bad information longer and so are less resilient in the face of integrity attacks.
- Matrix Organizations, with their cross-functional teams, may be able to overcome knowledge gaps caused by cyber-attacks; but, are often to coherently finish any tasking once the attack has begun.

At a technical level, a key finding is that agent-based simulation modeling that employs “social” reasoning and so agents at both the individual and group level is a significant win as it enables increased accuracy, increased predictive capability in general, and increased speed/number of actors modelable. Traditionally in agent-based modeling as you improve the model by making the agents more cognitively accurate, or by making the networks more realistic the number of agents that could be modeled or the speed of the model runs decreased. Our results demonstrate that adding social cognition to the model and the associated multi-level actions actually increased the number of agents modelable or the speed for the same number of agents, in the cyber-security domain.

At a measurement level, this research led to a temporal approach and a set of network of metrics for assessing organizational resiliency to cyber attacks and other organizational issues – see Figure 2. The basic idea is that a metric of interest can be used to assess immediate impact, persistent impact and recovery by varying the time period of interest. These metrics take the into account the lines of authority and communication among personnel, access to data and IT systems, mode of communication, and direct IT to IT connections. Both static and dynamic metrics were developed. In general, we find that the dynamic metrics are more effective for assessing resiliency than the static.

Bibliography

- Andersen, David, Dawn M. Cappelli, Jose J. Gonzalez, Mohammad Mojtahedzadeh, Andrew P. Moore, Eliot Rich, Jose Maria Sarriegui, Timothy J. Shimeall, Jeffrey M. Stanton, Elise A. Weaver, and Aldo Zagonel. 2004. Preliminary System Dynamics Maps of the Insider Cyber-threat Problem. Paper read at System Dynamics Modeling for Information Security: An Invitational Group Modeling Workshop, 16-20 February, at Pittsburgh, PA, USA.
- Bigley, Gregory A. and Karlene H. Roberts. 2001. The incident command system: High-reliability organization for complex and volatile task environments. *Academy of Management Journal*, 44, 6, 1281-1300.
- Carley & Svoboda, 1996. Kathleen M. Carley & David M. Svoboda, 1996, Modeling Organizational Adaptation as a Simulated Annealing Process. *Sociological Methods and Research*, 25(1): 138-168
- Carley, Kathleen M., 1990, "Group Stability: A Socio-Cognitive Approach," *Advances in Group Processes: Theory and Research*. Edited by Lawler E., Markovsky B., Ridgeway C. and Walker H. (Eds.), Vol. VII. Greenwich, CN: JAI Press, 7: 1-44.
- Carley, Kathleen M., Michael K. Martin and Brian Hirshman, 2009, "The Etiology of Social Change," *Topics in Cognitive Science*, 1.4:621-650.
- Zhiang Lin and Kathleen M. Carley, 2003, *Designing Stress Resistant Organizations: Computational Theorizing and Crisis Applications*, Boston, MA: Kluwer.
- Cheung, S.; Lindqvist, U.; Fong, M.W.; , "Modeling multistep cyber attacks for scenario recognition," DARPA Information Survivability Conference and Exposition, 2003. Proceedings , vol.1, no., pp. 284-292 vol.1, 22-24 April 2003
- Chi, M.T.H., Feltovich, P.J., & Glaser, R. , 1981. Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5, 121-152.
- Dennis K. Nilsson and Ulf E. Larson. 2008. Conducting forensic investigations of cyber attacks on automobile in-vehicle networks. In *Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, and multimedia and workshop (e-Forensics '08)*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), ICST, Brussels, Belgium
- Gao, Zhiqiang and Ansari, N., 2005 , "Tracing cyber attacks from the practical perspective," *Communications Magazine, IEEE* , vol.43, no.5, pp. 123- 131, May 2005
- Harrison, J.R. and G.R. Carroll. 1991. Keeping the Faith: A Model of Cultural Transmission in Formal Organizations. *Administrative Science Quarterly*, 36, 552-582.
- Kameswari Kotapati, Peng Liu, Yan Sun and Thomas F. LaPorta, 2005, A Taxonomy of Cyber Attacks on 3G Networks Intelligence and Security Informatics Lecture Notes in Computer Science, Volume 3495/2005, 129-138

- Klein, G. A., & Hoffman, R. R., 1993. Seeing the invisible: Perceptual-cognitive aspects of expertise. In Rabinowitz, M. (ed.), Cognitive science foundations of instruction. Hillsdale, NJ: Erlbaum. 203-226.
- Klein, Gary A. 1998, "Sources of Power: How People Make Decisions", MIT Press, Cambridge, Mass, pp. 1-30.
- La Porte, Todd R. and Paula M. Consolini. 1991. Working in Practice But Not in Theory: Theoretical Challenges of 'High-Reliability Organizations'. Journal of Public Administrative Research and Theory, 1, 1, 19-47.
- Lala, C. and B. Panda, 2001, Evaluating damage from cyber attacks: a model and analysis, IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 31(4):300-310.
- Lant, T.L. and S.J. Mezias, 1992, "An Organizational Learning Model of Convergence and Reorientation," Organization Science, 3(1): 47-71.
- Lee, Earl E. II, John E. Mitchell, and William A. Wallace. 2007. Restoration of Services in Interdependent Infrastructure Systems: A Network Flows Approach. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 37 (6):1303-1317.
- Li, Muhai, Ming Li and Xiuying Jiang, 2008, DDoS attacks detection model and its application, WSEAS Transactions on Computers, 7(8).
- Lipson, Howard F., 2002, Tracking and Tracing Cyber-Attacks: Technical Challenges and Global Policy Issues, Technical Report, Software Engineering Institute, Carnegie Mellon University
- Roberts, K. H. (1990). Some Characteristics of High-Reliability Organizations. Organization Science, 1, 160-177.
- Scherrer, Joseph H., and William C. Grund. 2009. A Cyberspace Command and Control Model. edited by U. A. Force. Maxwell AFB, AL: Air War College.
- Schulman, P. R. (2004). General attributes of safe organizations. Quality and Safety in Health Care. 13, Supplement II, ii39-ii44.
- Stytz, Martin & Sheila Banks, 2008, Advancing Cyber Warfare Simulation System Capabilities, SimTecT 2008 Simulation Conference: Simulation - Maximising Organisational Benefits (SimTecT 2008) Melbourne, Australia, May 12 – 15 , 2008
- Weick, K. E., & Roberts, K. H. (1993). Collective Mind in Organizations: Heedful Interrelating on Flight Decks. Administrative Science Quarterly, 38, 357-381.

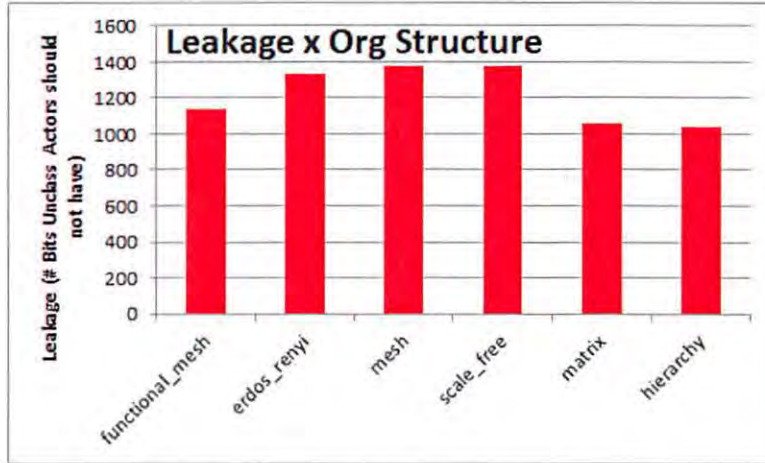
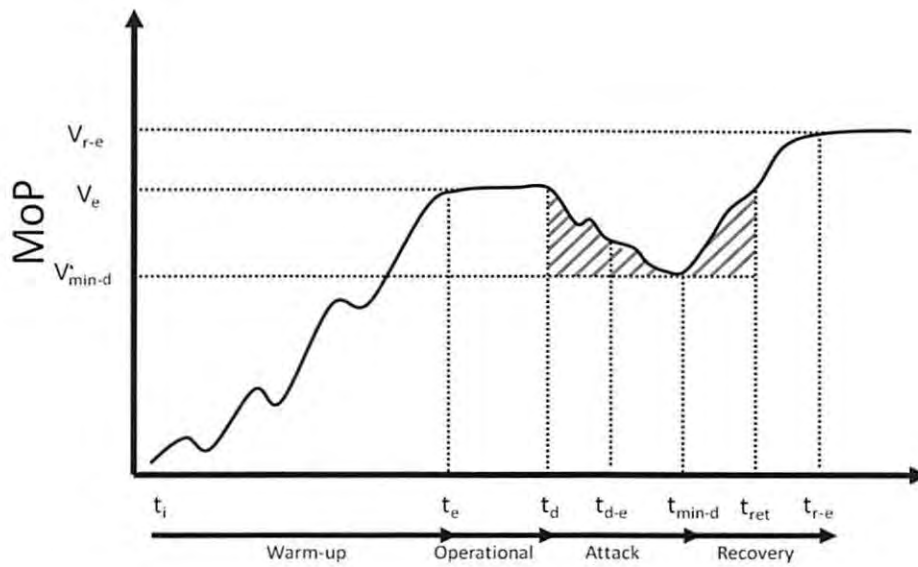


Figure 1. Inadvertent Leaks under cyber-attack.



Impact:

Immediate Impact: $I_m = V_e - V_{min-d}$

Persistent Impact: $I_p = V_e - V_{r-e}$

Recovery:

Recovery Time: $t_{r-e} - t_{min-d}$

Figure 2. Measuring resilience from a temporal perspective

Security Behavior Observatory: Infrastructure for Long-term Monitoring of Client Machines

Alain Forget, Saranga Komanduri, Alessandro Acquisti,
Nicolas Christin, Lorrie Faith Cranor, and Rahul Telang

July 14, 2014

[CMU-CyLab-14-009](#)

[CyLab](#)
Carnegie Mellon University
Pittsburgh, PA 15213

Security Behavior Observatory: Infrastructure for Long-term Monitoring of Client Machines

Alain Forget^a, Saranga Komanduri^b,
Alessandro Acquisti^c, Nicolas Christin^d, Lorrie Faith Cranor^e, Rahul Telang^f
Carnegie Mellon University

^aforget@cmu.edu, ^bsarangak@cs.cmu.edu,

^cacquisti@andrew.cmu.edu, ^dnicolasc@cmu.edu, ^elorrie@cmu.edu, ^frtelang@andrew.cmu.edu

Abstract—Much of the data researchers usually collect about users’ privacy and security behavior comes from short-term studies and focuses on specific, narrow activities. We present a design architecture for the Security Behavior Observatory (SBO), a client-server infrastructure designed to collect a wide array of data on user and computer behavior from a panel of hundreds of participants over several years. The SBO infrastructure had to be carefully designed to fulfill several requirements. First, the SBO must scale with the desired length, breadth, and depth of data collection. Second, we must take extraordinary care to ensure the security and privacy of the collected data, which will inevitably include intimate details about our participants’ behavior. Third, the SBO must serve our research interests, which will inevitably change over the course of the study, as collected data is analyzed, interpreted, and suggest further lines of inquiry. We describe in detail the SBO infrastructure, its secure data collection methods, the benefits of our design and implementation, as well as the hurdles and tradeoffs to consider when designing such a data collection system.

I. INTRODUCTION

Our understanding of the security and privacy challenges users face has grown substantially since some seminal usable security papers were first published [1], [2]. Much of the empirical data relating to topics such as authentication [3], [4], computer warnings [5], phishing [6], [7], identity theft [8], has been collected through either in-lab or online controlled experiments, or with surveys and interviews. Controlled lab and online studies allow researchers to isolate variables to observe and measure specific phenomena and effects. Survey and interview data have given us a better understanding of users’ perceptions and perspectives, which are invaluable if we are to make security and privacy systems more usable. However, lab studies often lack ecological validity, since users may behave differently in the real world than in an artificial experimental setting [9]. Furthermore, self-reported data may not match users’ actual behavior [10], [11].

Thus, the research community has begun focusing on more ecologically-valid data collection. Most published field studies to date have concentrated on specific sub-areas in the usable security and privacy field (e.g., text passwords [12], [13], ATM usage [11], malware infection [14], [15], mobile locking [16], social networks [17]). Most of these studies have short-term focus and monitor only a specific aspect of user or machine behavior. If we are to discover the ground truth of users’ most pressing security and privacy challenges, it seems important to

collect data on users’ and their computers’ overall naturalistic behavior in the wild over an extended period of time.

In this paper, we present and describe the Security Behavior Observatory (SBO) we designed to help researchers collect more ecologically-valid data of the widest possible scope over several years. The SBO is a client-server infrastructure for collecting data from a panel of several hundred household computers. Our software will allow us to deploy modular and independent sensors to monitor many security and privacy aspects of home computer use. Observing comprehensive and real-time decision-making of a large panel of users over an extended period of time in a real world setting, in itself, is invaluable. This information can provide a variety of practical and powerful insights into improving security and privacy policies and technologies. However, designing and building the SBO requires attention to factors less frequently considered in shorter-term, more focused studies. The infrastructure must be sufficiently scalable, reliable, and robust to collect the required size, breadth, and depth of data over the study’s lengthy duration. In addition, we must carefully consider how best to maintain the security and privacy of participants’ data, given the sheer amount and detail of behavioral data we will collect. We also require the flexibility to adjust the types of data we collect throughout the study, since research needs will invariably change as earlier data analysis leads to further lines of inquiry.

This paper is organized as follows. Section II describes how this project contributes to the science of security. Section III introduces the SBO and provides examples of the data we intend to begin collecting. Section IV elaborates on the SBO’s architecture from two perspectives. First, we use a data flow model (Figure 1) to describe how data is collected from participants’ client machines and sent to our server, and describe the specific benefits of our design decisions. Second, we use a deployment model (Figure 2) to describe our server configuration and how it securely and reliably handles the data encryption, transfer, and storage procedures. We briefly describe how participants enroll in our study in Section V. Section VI discusses some challenges, trade-offs, and limitations to consider when designing and deploying such an SBO system. Finally, we describe related work of similar data collection endeavors in Section VII and offer some concluding remarks in Section VIII.

II. THE SCIENCE

Our understanding of computer and user behavior, with respect to security and privacy, has largely been based on studies of short duration and narrow focus. These studies have helped guide research over the past 20 years. However, a large-scale field study permits the measurement of users' security and privacy challenges and behaviors with much greater ecological validity than in the lab, where the experimental setting might not reflect users' actual behavior in their natural environment [?]. Furthermore, a long-term longitudinal study would provide data on the frequencies at which users encounter various security and privacy issues. These frequencies would represent risk probabilities, which are a key element of any risk assessment or risk management strategy. Thus, data from such a field study could be used to both inform and prioritize future research agendas.

To fill this need for more ecologically-valid data, we have built the Security Behavior Observatory (SBO): a framework for collecting data from a large panel of end-users whose online behavior will be monitored and analyzed over an extended period of time. This project is now possible thanks to widespread access to broadband Internet connections with reasonable upload speeds. The SBO offers an unprecedented window on real-time, real-life security and privacy behavior in the wild. Through the SBO, we aim to contribute to the evolution of a data-driven science of information security, with immediate applications in usability, economics, and secure system design. We hope this project will encourage discussions on collecting ecologically-valid data in current research practices, and serve as a template for future field studies.

III. SECURITY BEHAVIOR OBSERVATORY

The Security Behavior Observatory (SBO) is a client-server architecture where participants' client computers are monitored over an extended period of time and upload collected user and computer behavior data to our servers. The initial launch of the SBO will monitor computers running Windows Vista, 7, and 8. We currently focus on these operating systems because their underlying architectures are almost identical (at least for the purposes of data collection), Windows has been the most popular operating system for the past 5 years [18], and desktop usage remains dominant over mobile computing [19]. However, the high-level infrastructure design and our own implementation (both described throughout this paper) can easily be applied to other operating systems (see Section IV-A7). Examples of the data we intend to monitor from hundreds of client machines over several years, with IRB approval and under strict security and privacy safeguards, include those described in the following subsections.

Our architecture is designed to provide data covering as much of the security and privacy space as possible. Some example research questions we intend to examine include:

- How up-to-date are operating systems?
- How long before a clean machine is infected, and how does infection actually occur in the wild?

- What are users' online social network privacy settings? Do they ever change, and why?
- What warning dialog messages do users encounter most often, and how do users respond?

As of this writing, we are performing final tests on most of the implemented data collection sensors (see Sections III-A to III-E) while the others are under development (see Sections III-F to III-H). We intend to invite participants to complete questionnaires and interviews to elicit their perspectives on issues and events we observe throughout the study. We are beginning a pilot study on the main client-server SBO infrastructure (see Sections III-I to IV-A) and user study methodology (see Section V). We have purchased the server deployment configuration (see Section IV-B) and hope to begin data collection no later than this summer.

A. Filesystem

As currently designed, the SBO tracks changes to the filesystem, including the added, modified, or deleted file's size, last date modified, permissions, and other related information.¹ This data will help determine, for instance, if malware exists on the system and if so, how it affects machines' file systems, and whether or not users are likely to have noticed its presence.

B. Installed software and operating system updates

The SBO maintains a list of installed applications, their version numbers, and other related data, to determine what privacy or security software (e.g., anti-virus, firewall, ad-blockers, anonymizers) are installed, and whether they are up to date. The SBO also tracks which (and how soon after their release) operating system updates and patches have been installed. This allows us to measure the duration and severity of client machines' vulnerability to security threats.

C. Processes

The SBO monitors which processes (e.g., programs, applications) are running on clients' machines. It captures when all processes start and terminate, and can provide additional process status information at regular intervals. Primarily, this data will assist with the detection of malware. The SBO also collects general computer usage statistics that may help prioritize future security and privacy work, such as towards frequently-used applications.

D. Security-related events

The SBO also notes general security-related events, such as account-related events (e.g., logins, settings changes, password changes), registry modifications, wireless network authentications, firewall changes, and potential attacks detected by the operating system. This will provide valuable insights on multiple usable security topics, including the security measures users' employ on their computers, potentially dangerous program behavior, and the types and frequency of attacks that occur on home users' machines.

¹However, we do not collect file or network packet contents since this may be too invasive and bandwidth intensive.

E. Network traffic

The SBO captures all network packet headers sent and received to clients' computers.¹ This data would allow us to detect various network traffic types that may be risky (e.g., peer-to-peer file transfers, dangerous websites) or suspicious (e.g., malware, intrusion attacks). We could thereby verify whether risky Internet behavior is correlated with a higher probability of an attack or infection.

F. Internet browsing behavior

We intend to further monitor users' web browsing behavior by collecting data from Microsoft Internet Explorer, Mozilla Firefox, and Google Chrome. We intend to capture search queries, online social network activity, browsers' and some online accounts' privacy and security settings, as well as other behavior of particular research interest (e.g., social networks, behavioral advertising). One example of possible analyses includes: what are users' privacy settings and behaviors on online social networks, do said settings adequately preserve users privacy, and if not, how could the website be better designed to empower users to more easily and accurately express their desired privacy settings. Another example of planned analysis consists of measuring how often users' actually make purchases derived from behavioral advertising links. This would reveal insights on the actual utility users gain from behavioral advertising, with respect to the privacy cost.

G. Configuration of software and online accounts

We also intend to track the security and privacy settings of users' software (see Section III-B) and online accounts (e.g., Facebook, Twitter). This would provide data regarding users' security and privacy practices. Should users change any such settings during the course of the study, it will be particularly interesting to understand users' motivation for initiating the change. If this could not be inferred with our data (i.e., if we did not detect any particular event preceding the setting modifications), we may send participants a survey or request an interview to inquire further.

H. Warnings

We intend to capture the content of and users' response to warning dialogs that request users make a security- or privacy-related decision. Past research has shown that users frequently do not understand these warnings, let alone know how to respond [5], [20]. This data would bring insights into the warnings users must cope with most frequently and what security and privacy decisions users make when prompted.

I. Security, Privacy, Usability, and Research Requirements

To capture such a wide array of data types over a long period of time, it is crucial we design and build an infrastructure that satisfies several requirements. First, we should minimize the impact of our data collection software on participants' computing and network performance. Thus, since the amount of data we can gather and transmit from clients is limited, we need the ability to be selective with and vary the types

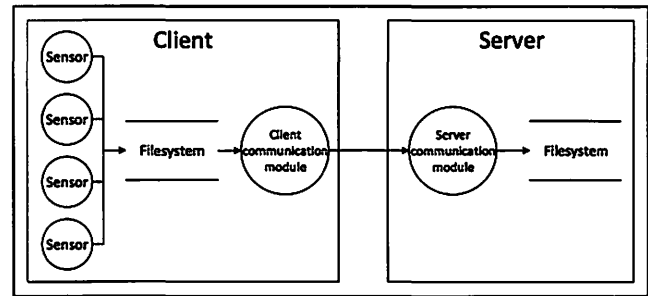


Fig. 1. Data flow between our SBO client and server software.

of data we collect over time. Second, as we collect and analyze data, we expect our research questions will evolve and require different types of data to be answered. For these reasons, our data collection architecture must be flexible enough to accommodate our changing needs. Third, unlike most experimental software which is typically used for only a short time for specific targeted purposes and environments, any problems caused by our client software could profoundly impact participants' computing experience, due to the breadth, depth, and duration of our data collection. Thus, our system requires a much higher degree of stability and reliability than typical experimental software. With these requirements in mind, we have designed and implemented the following architecture for the SBO.

IV. ARCHITECTURE

In this section we describe our design and implementation of the SBO architecture from two perspectives. We first illustrate how the data flows from initial collection on the client to storage on our server. Second, we discuss our deployment of servers and each of their roles. For both of these perspectives, we highlight the specific benefits of our design.

A. Data Collection and Flow

Figure 1 shows a data flow diagram of the client-server architecture. Each type of data is collected by a *sensor*, which outputs the data into a common directory. The *client communication module* periodically checks this directory for data files, and compresses, encrypts, and sends them over an SSL-encrypted channel to the *server communication module*. This architecture provides a number of beneficial design features.

1) *Silent updates*: We use Windows Installer [21] to package all the client software components into a single executable. Windows Installer provides functionality for cleanly installing and uninstalling the software, as well as upgrading. When the client communication module establishes a connection to the server, it first verifies that client software is up to date. If the server determines that it is not, the server provides a link to the current version's installation executable (hosted on our server) to the client. The client then disconnects from the server, downloads the current version of the client software, and checks the file's integrity with an MD5 hash. If the file is intact, the client shuts itself down after silently running

the installer executable in the background. Windows Installer then performs a “major upgrade” whereby the previous version is completely uninstalled before installing the new version. This clean-install approach avoids potential complex problems that can occur with minor upgrades and patches, which can result in an unstable software state. Should the update fail for some reason, Windows Installer will roll back to the previous software version, and the data collection can continue until the client attempts the update again. The entire update process is completely invisible to the user, and does not affect their normal computer usage in any way.

2) *Independent sensors*: Each type of data of interest (see Section III) is collected by a software *sensor* we have designed and implemented. Each sensor is independent of the rest of the data collection system. This sensor independence provides the following robustness and adaptability benefits. Firstly, if a sensor fails, the other sensors will continue to collect data, which the client communication module will continue to upload to the server. Secondly, if the client communication module fails or the server is unavailable, the client sensors will continue to collect and store data locally, and upload the data once the client communication module has finished restarting and/or the server becomes available. Thirdly, as the data interests for the study change over time, sensors can be silently (see Section IV-A1) and independently added, enabled, configured, disabled, or removed by the experimenters at any time without impacting any other aspect of the client system or our software. Finally, sensors can be implemented in whichever language is best for collecting the desired data. In Windows, this is most often a .NET language (e.g., C#, PowerShell), a command-line batch script, or Java.

3) *Least privilege*: To ensure clients’ security and privacy, the principle of least privilege should be followed whenever possible. However, some data we seek to collect is likely to require administrator access to the client system. Fortunately, our architecture’s sensors are independent, so higher privileges can be given only to sensors that require them.

4) *Minimal footprint*: Since the study’s primary goal is to *observe* computer users’ typical behavior, we must take care to avoid experimental effects that may influence this behavior. Thus, users should not notice a decrease in computer or network performance during the study. We achieve this in two ways. First, we take care to avoid intensive processing or blocking access to system resources as much as possible. Second, we throttle our client software’s data upload speed to at most 192 kilobits per second (kbps), which is half of the slowest upload speed of the least expensive home Internet service plan available (excluding dial-up) in our initial area of participant recruitment (see Section VI-D). A data transfer rate of 192 kbps is equivalent to about 1.44 megabytes per minute, which is not much bandwidth for on-going data collection. This further enforces a minimal footprint by requiring the experimenters to be selective about what types and richness of data we collect. Although necessary, prioritizing what data to collect can be challenging (see Section VI).

5) *Minimal user interaction*: The use of passive observation to avoid experimental effects also implies we must minimize any user interaction. Our sensors and client communication module execute as Windows *services* [22], which implicitly provides this benefit. A Windows service is an executable program that runs in the background. Similar to Unix daemons, services (or any process or thread they spawn) cannot display any form of user interface (since Windows Vista). Thus, should a program running as a service attempt to display anything to the user, it will not be shown. This acts as a safeguard to ensure that we do not influence the user’s normal computing tasks. However, this can be a challenge should the experimenters purposefully desire to interact with the user. This may be desirable should the experimenters wish to test participants’ behavior to some stimuli. If future research questions require this, the application containing the stimuli would run as a standard program, not as a service, and would be designed so that any disruption to the user is minimized. However, the stimuli, and any effects it may have on all data being collected, should be carefully considered.

6) *Multiple user accounts*: Participants’ computers may have multiple accounts. The computer’s owner may have a separate account for guests, or each member of a household may have a separate account on a common machine. It is crucial that our data collection software run regardless of which user account may be logged in. Fortunately, Windows services can be set to always run when the system starts, independently of which user(s) logs in or logs out. Since our sensors and client communication module run as services, they are assured to run irrespective of which users login. Standard (non-service) applications can also be executed at startup, regardless of which user logs in, by adding a value to the registry [23].

7) *Portability*: Although we are currently targeting only Windows machines, we may desire the flexibility to collect data from other operating systems (OSes). To do so, we would almost certainly need to write new sensors, since the Windows underlying architecture is completely different from Unix-based operating systems. However, the client and server communication modules are written in Java, and thus should be easily-portable to any OS.

B. Deployment

There are several high-level requirements the SBO must meet. It is crucial that the data is securely and efficiently collected from participants. The data must also be as securely and reliably stored as possible. Finally, researchers must be able to access and work with the data with as little inconvenience as possible. Figure 2 illustrates the deployment of our server architecture we believe best meets these requirements. We describe below each physical server’s role, how data flows from the clients to the various server machines, and the security precautions that are in effect throughout.

1) *Data collection server*: The data collection server’s role is solely to receive data from clients, and periodically send said



Fig. 2. Our SBO high-level hardware architecture and data flow.

data to the data analysis server when requested. The data flow from clients to the data collection server proceeds as follows:

- 1) Data is continuously generated on client machines (see Section IV-A).
- 2) At regular intervals, each client establishes an SSL connection to the data collection server.
- 3) The client and server mutually authenticate each other by encrypting random numbers with a shared symmetric authentication key [24].
- 4) When the server is ready to receive data, the client compresses the data, encrypts the compressed data with its symmetric encryption key (which is distinct from the authentication key and unknown to the data collection server), and sends it to the server.
- 5) The server stores the data locally, still encrypted with the client's encryption key.

2) *Data analysis server*: The purpose of the data analysis server is to periodically retrieve the encrypted data from the data collection server (and thereafter delete it from the data collection server), store all collected data in the data storage node(s), and provide access to researchers to perform work with the data. To ensure the data's security, it must remain solely on the data analysis server and be accessible only to project administrators and researchers. Thus, the data analysis server can be accessed only through a secure shell (SSH) tunnel originating from the specific IP addresses of the researchers' and administrators' work machines. To remotely access the data analysis server, researchers and administrators must first remotely connect to their work machine and, through said machine, establish an SSH tunnel into the data analysis server. Since the data must never exist anywhere other than our servers, all work with the data must be performed through this SSH tunnel.

As previously mentioned, the data analysis server periodically requests clients' encrypted data from the data collection server. This data transmission occurs over a mutually-authenticated SSL connection [24], and is scheduled to occur at a time of day when the data collection server is least likely to be busy receiving data from clients (e.g. 4:00 AM). The received data is still encrypted with the corresponding clients' symmetric encryption key (see Section IV-B1).

Clearly, the data cannot be analyzed while it is encrypted, but we also cannot risk storing it on the server unencrypted. Section VI-C discusses how we handle the decryption of the data for analysis.

3) *Data node(s)*: Participants' encrypted data is ultimately stored in two places; in the data storage node(s) and data backup node(s). The backup node(s) are located in a physically-separate building from the storage nodes. These nodes are accessible only through the data analysis server, which represents the storage and backup nodes each as a network-attached storage (NAS) ZFS volume [25], [26]. Some key features of ZFS include snapshots (i.e., simple revision control), error detection, protections against data corruption, and storage pools, which allow the single logical ZFS volume to dynamically expand to include additional physical volumes. Thus, as our needs for additional storage grow and we add storage nodes, the additional storage space can simply be added to the existing logical ZFS volume, rather than being represented as a new volume (which would require additional researcher effort to manage and organize the data among multiple logical volumes).

An alternative filesystem could be the Hadoop Distributed File System (HDFS) [27], [28]. With similar benefits as ZFS, HDFS also allows data processing and analysis to be parallelized by distributing the data and computations among the nodes to more quickly process the data. However, HDFS cannot be treated as a traditional logical volume; it must be accessed through a special interface (i.e., API). Furthermore, programs must be written in a particular way to leverage the parallelism benefits of HDFS. Thus, Hadoop may require significant investment costs of time and effort. Furthermore, Hadoop would be beneficial when there are several data storage nodes which can perform computations in parallel. However, we currently need only a single storage node with an 8-core CPU to begin data collection, so the parallelism gains are not worth the time and effort investment. As the size of our panel and collected data grows to require several storage nodes, we will consider using Hadoop or another data storage and management technology instead of ZFS.

V. USER STUDY METHODOLOGY

With the aforementioned infrastructure in place and having already obtained approval from our institutional review board for these procedures, we can solicit users to participate in our panel. Our primary method of finding participants is through a recruitment service for which people have asked to be notified about experiments. Potential participants will be asked a number of pre-screening questions. Participants must be over 18 and own a Windows Vista, 7, or 8 personal computer. We send interested persons an e-mail with a link to where they can complete the following initial enrollment tasks:

- 1) Reading and completing a consent form, which clearly informs users that we may monitor all activity on their computer and collect any data except for the contents of personal files, e-mails sent or received, content of documents on Google Docs, and bank card numbers.

- 2) Providing the names and e-mail addresses of others who also use the computer to be instrumented, so we can obtain their consent.
- 3) Completing an initial questionnaire
- 4) Download and install our data collection client software

Once these steps are complete, and all the other users of the computer have provided their consent, the participant is awarded a \$30 Amazon.com gift card, since we can now collect data from the participant's machine. Participants thereafter receive a \$10 gift card for every month our client software continues to upload data from their computer. This data transmission occurs silently in the background without requiring any action from participants. We also send periodic e-mails informing participants that either everything is working fine, which of the above enrollment tasks still need to be completed, or if we are not receiving data from their machine. If we do not receive data from users for 3 months, we may cease their participation.

VI. DISCUSSION

There are a number of issues warranting careful consideration when collecting data from hundreds of participants' personal machines.

A. Participant IDs

It is necessary for our server to be able to identify which client belongs to which participant for several reasons. Primarily, every client machine must locally store its unique encryption and authentication keys to encrypt its data and securely communicate with our server (see Section IV-B1). We also need to verify users' continued participation (i.e. uploading data), so we can compensate them or remind them that they need to keep their computer on and connected to the Internet to continue participating. Additionally, we wish to be able to perform participant-specific data analyses to evaluate whether particular demographics are correlated with certain behaviors. We also wish to perform longitudinal analyses across specific machines' lifetimes (e.g. time before a malware infection).

The easiest way to identify client machines is to prompt the user for their assigned ID when they first install our client software. However, because our software runs as a Windows service (see Section IV-A5), it cannot display any user interface elements, and thus cannot interact with the participant. We solved this problem by creating an independent program that verifies that the stored participant ID and keys are valid, and if they are not, the program prompts the user. This program is run as a standard process, independently of any of our services, which allows it to interact with users if necessary. However, since the program does not run as a service, it does not execute within the same workspace or with the same privileges as the rest of the client software. Thus, we had to resolve various challenges regarding program-service communication, differing access control privileges, and synchronization.

B. Ethics & participant privacy

Although true for all user studies, it is critical that an institutional review board (IRB) approve the study's methodologies and procedures to ensure participants' are treated ethically and their data is kept confidential and secure. We spent considerable time iterating over our consent procedures with our IRB before their approval. However, many review boards do not have the expertise to understand the specific security and privacy challenges that may arise. Thus, the burden lies on the experimenters to consider carefully which data they are willing to collect and hold in trust, and to weigh the risk of a compromise with the value of such data to the advancement of the community's knowledge. Regarding de-identification, participants are assigned a random ID, which decouples their uploaded data from their provided personal information. We are also considering additional anonymization strategies and weighing their costs (e.g., loss of data richness, client-side computational loads) against possible threat models (e.g., client, network, server attacks).

C. Data Security

Given the potential sensitivity of the data our infrastructure collects and transmits from client machines across the Internet and stores on our servers, the data's security and confidentiality must be carefully considered and strictly enforced. In our implementation, we employ reliable end-to-end data encryption. Every client is assigned a unique encryption key. Client-side keys are stored in a permission-secured file on the client. To obtain the keyfile, an attacker would need access to the client with elevated privileges. The value of a participant's keys is unclear in this scenario, since this attacker could install malware to collect more sensitive information (e.g., passwords, bank account numbers) than we do.

Before transmitting the data, the client communication module compresses and encrypts the data with 128-bit AES [29] using Cipher Block Chaining mode [30] and PKCS5 Padding [31]. This encrypted data is sent to the server through an SSL connection and stored, still encrypted with the client's unique key. Once the encrypted data is received by the server, the client-side copy is deleted.

Although methods for computing on encrypted data exist (e.g., homomorphic encryption), our analyses across multiple sensors' data longitudinally across time are likely to be complex enough that they would not be practically feasible with such solutions. Instead, one researcher with access to all the clients' keys (stored in an isolated and secured MySQL database owned by a separate, dedicated, and tightly-secured user account) will decrypt and decompress each client's data into a TrueCrypt volume, to which all project researchers will have the key to analyze the data. Unencrypted data may temporarily exist in memory while and after working with it. However, the data must remain on the data storage nodes, which can be accessed only through a secure shell to the data analysis server from the specific IP addresses of the researchers' own campus machines. No other connections to this server are permitted. We feel that this is the best solution

for offering sufficient data security without overburdening researchers with complex, time-consuming procedures to access the data.

D. Client upload bandwidth

Given the wide breadth and depth of data we ideally wish to collect, the limit on how much data we can realistically collect is the clients' upload data rate. We must restrict the amount of data we upload from participants' machines to avoid a noticeable reduction of their Internet connection bandwidth.

To calculate this maximum upload rate, we first found the lowest data upload rate of Internet plans in our area, which is 384 kbps (kilobits per second). To avoid a noticeable impact on participants' network performance, we should use only a fraction of this total upload rate. By using only half, the actual data rate our software should be allowed to use is 192 kbps, or 24 KBps (kilobytes per second).

To ensure we do not surpass our desired bandwidth usable, we throttle clients' data upload speed by interleaving data transmission and sleep commands. For example, to achieve an upload rate of 24 KBps, the client process could alternate between uploading 6 KB and then sleeping for 250 ms until all the data is transferred. Sleeping between data transmissions should cause the OS to flush the uploaded data stream (i.e., actually send the data to our server rather than leave it in the client's network buffer in case our process adds more data to be sent) and free the network bandwidth and processing cycles for other applications until our application resumes. We hope to add adaptive throttling functionality to upload either more data when the network and computer are idle or less when the machine and Internet are in heavy use. However, this risks biasing the lower-priority types of data that would be collected only for clients with more computing capability and network bandwidth, which might be higher-income participants.

Given the massive amount of data this infrastructure can collect about client machine behavior (see Section III), it is important to employ techniques to minimize the physical size of the data transferred and stored. One such technique involves sensors that perform periodic snapshots, which should only log differences between the previously-recorded and current state, rather than always logging the complete current state. This is particularly important for snapshot sensors that gather large amounts of data for every snapshot. For example, when monitoring the filesystem, we may want to know all files' permissions, size, date first created and last modified, and potentially an MD5 hash. Such a complete list could easily be at least several hundred megabytes in size, which is an unreasonable amount of data to regularly transfer and store. However, only logging differences between the previous and current snapshot would likely be a realistically manageable size.

Another technique we use to reduce our data logs' footprint is the Binary JSON (BSON) data format [32]. The BSON data format is ideal for our infrastructure's purpose, since BSON is specifically designed to minimize spatial overhead in data transfers and storage, be easily traversable, and be efficient to

encode and decode. We use BSON to log data that is either hierarchical in nature or may contain variable data elements (i.e., where there could be many null values if the data were logged in a flat table structure).

Even with data minimization techniques such as these, we anticipate having to make difficult decisions about which types of data to prioritize. However, while performing our preliminary data analysis, we may observe phenomena that we wish to further explore, but may be unable to because we had previously chosen not to enable the relevant sensors. To illustrate, suppose we initially choose to focus on malware infection. Thus, the minimum data we would need to collect appears to be network packet traffic, filesystem changes, and the executing processes. However, there are a number of scenarios where we would be missing data. For example, the network packet sensor would be unable to detect malware downloaded through SSL. Without the warning dialog sensor, we would not know if the user was ever warned from visiting a website, or prompted to download or install the malware. Without tracking security-related events, we may be unable to detect changes to the Windows firewall or other computer security settings. Admittedly, it may be possible to make some inferences from the sensors we did enable, but our understanding of the malware-infection events would certainly be incomplete. However, since we cannot possibly collect all the data, it is clear that there will be some limitations to the analysis we will be able to perform. Still, the choices of which data to collect in tandem will need to be made carefully, since poor decisions could pose unnecessary additional challenges when analyzing the data.

E. Server specifications & other cost considerations

There are a number of significant costs involved in conducting such a long-term data collection study. First and foremost, as discussed in Section IV-B, at least four physical server machines are required to begin the study; data collection, analysis, storage, and backup. Each of these machines have different specification requirements which should be carefully considered before committing to a purchase. The importance we place on each server's components are noted in Table I. Our reasoning for these priorities is as follows. The data collection server would benefit from several processor cores for receiving data from multiple clients at once, but these do not necessarily need to be the highest-possible clock speed (hence the *medium* rating). Our server software does not require much memory. This server's storage requirements are also relatively little, since it needs to retain only data collected over a few days, in case the data analysis server is temporarily delayed from removing the data (see Section IV-B2). The data analysis server itself also requires relatively little storage space for the operating system, data transfer, manipulation, and analysis applications and scripts. However, the data analysis server does require significant memory and processing power for its namesake purpose. The data storage nodes require at least a reasonably powerful processor and memory for data transfers to occur rapidly (and to quickly perform data processing tasks

TABLE I
IMPORTANCE OF EACH COMPONENT FOR EACH SERVER (SEE
SECTION IV-B).

Server	Processor	Memory	Storage
Data Collection	Medium	Low	Low
Data Analysis	High	High	Low
Data Storage & Backup	Medium	Medium	High

if HDFS is in use). Of course, the storage nodes must have sufficient space to hold all the data to be collected. We use the following calculation to estimate our long-term storage needs. Assuming each participant uploads approximately 24 KBps (kilobytes per second) to our server (see Section VI-D), this equates to 377.4 gigabytes (GB) per year per participant. In our study's first year, we intend to have 100 users participating in our study. Thus, 100 participants each generating 377.4 GB per a year results in about 37.74 terabytes (TB) of data. We have obtained a minimum hardware configuration that satisfies the above requirements, and can be expanded for further data collection beyond one year, for around \$35,000 (USD).

In addition to the server configuration costs, there are also on-going costs to be budgeted. Primarily, the participants require compensation. We are currently offering \$30 for completing the necessary initial tasks to begin participating in the study (see Section V), and \$10 for every month they continue to participate (e.g. we continue to regularly receive data from their machine). These costs add up quickly, since each participant costs \$150 per year, so 100 participants cost \$15,000 for a single year. Furthermore, if we cannot initially attract enough participants, we may need to consider increasing this stipend, which would further increase costs. Other on-going costs that should be considered include the technical administration and maintenance of the server hardware as well as at least one dedicated project leader (and ideally a support team) to build and continuously refine the software and sensors, oversee the smooth execution of the study, and lead the data management (see Section VI-C) and analysis.

F. Study Limitations

Despite the wide scope of this infrastructure and study, there are some limitations which must be noted. Firstly, we are currently targeting only participants using Windows Vista, 7, or 8. Our focus on modern Microsoft operating systems (OS) means that we may not observe phenomena that occur on Unix-based OSes. Furthermore, mobile devices and tablets are growing in popularity [19]. Users' behavior and risk with respect to privacy and security with these devices may differ significantly than with traditional desktops or laptops. For future work, we could build sensors to collect data on Unix-based systems' usage, as well as mobile devices and tablets. Fortunately, our client communication module (see Section IV-A) can run on any system that supports Java (which includes most modern operating systems, see Section IV-A7).

In our user study, we ask users to install our software only on their one main Windows computer, because we are interested in observing the breadth of behaviors of multiple

independent machines. However, people often have multiple devices through which they may have privacy and security challenges, including mobile devices and tablets. Thus, a complete in-depth examination of participants' behavior would require instrumenting all of a user's devices. This would be particularly challenging, given the multiple OS architectures participants may use. It is also unclear whether or not a participant's work machines should be instrumented. This would be required for a truly complete understanding of users' computing experience and behavior, but it would require participants' employers' consent, since data collection software on these machines may unintentionally capture the employers' intellectual property or other sensitive data. In any case, as our user study is currently designed, even though we capture a wider breadth of data than previous studies, we still risk missing some behaviors that occur on participants' non-instrumented devices. In future work, we hope to also collect data from mobile devices and tablets. We hope to reuse our client communication module to collect data from devices that support Java (see Section IV-A7).

As previously mentioned (see Section V), we offer participants \$30 to complete the initial enrollment, and \$10 per month of continued participation. This may bias our sample towards lower-income and privacy unaware or unconcerned participants. We will be able to confirm the former by asking participants to self-disclose their income in our enrollment questionnaire. However, it is unclear if any affordable level of compensation could attract higher-income participants. Additional compensation may also fail to attract privacy-concerned users, since users willing to be monitored are likely to do so for relatively small immediate short-term gains [33], [34].

VII. RELATED WORK

Lalonde Lévesque et al. [14] performed a 50-subject 4-month study of the effectiveness of an anti-virus software (AV) with respect users' computer behavior. Participants were given a Windows 7 laptop with Trend Micro's premium home anti-virus software and various monitoring software and scripts pre-installed. Every month, participants were required to meet with the experimenters to complete a survey about their computer usage and for the data to be collected from the machines. The AV detected 95 distinct threats on 38% of machines during the study, the vast majority of which were trojans, which is comparable with publicly-available statistics [14]. The authors' found 18 threats (e.g., 7 unwanted software, 9 adware, one malware, and another suspected as malware) that the AV failed to detect on 20% of machines. Participants with a greater computer expertise were more at risk of being exposed to threats than less computer-knowledgeable users. Furthermore, the authors reported that visiting sports and Internet infrastructure sites were more associated with a higher rate of infection, while visiting sites with pornographic or questionable content was less so. Although their methodology bares some resemblance to ours, there are several important differences between this and our study. Most obviously, our target sample size and study duration will both be several times

greater (i.e., hundreds of participants over several years). A more fundamental difference lies in our respective experimental models. Their study follows a “clinical trials” experimental model from medical research, whereby subjects are given a *treatment* (i.e., AV) and its effects are monitored over time. In contrast, our study’s primary purpose is to passively observe our participants’ and their machines’ behavior by collecting a very wide array of security- and privacy-related data (see Section III) without any form of experimental intervention whatsoever.

Van Bruggen et al. [16] instrumented 149 student participants’ Android smartphones with software that collected two types of data over two weeks; usage statistics (e.g., data usage, text messages, screen lock) and participant responses to weekly surveys on various topics. They found that 65% of their participants used a phone locking mechanism; 51% used the Android pattern lock and 14% chose a text password or PIN. They found no correlations for this choice with gender, previous phone type, text message frequency, data usage, or personality traits. Upon being surveyed about their password sharing behavior, 19% responded that they shared the password to their phone, while 63% shared passwords for other devices or services. The authors suggested that participants may place greater value the security of the mobile device over other devices or services. The authors later employed intervention messages based on incentives, morality, and deterrence to encourage users to either adopt a screen lock or upgrade to a more secure lock (e.g., from the pattern lock to a text password). The interventions did not appear result in many conversions. The authors concluded that the cost associated with targeting the users and implementing the interventions may not be worth the limited results. Our study does not currently target smartphones or attempt to modify users’ normal computing behavior, we may consider testing attempts to assist, inform, and persuade users to take security precautions, should our data suggest that many users leave their computers dangerously vulnerable or otherwise behave insecurely. We also hope to expand our study in the future to include a broader range of devices, including smartphones and tablets.

Florêncio and Herley [12] collected Internet password data from over a half-million people over 85 days. This data was collected voluntarily from users of the Windows Live Toolbar. Their component hashed and stored passwords users’ entered in web pages’ password input fields, as well as the related URL, the passwords’ bit strength, and other data. The authors also tracked incidents of password re-use as follows. Every time a character was typed into the web browser, their system hashed and compared each sequence of the last 7 to 16 typed characters to each of the stored password hashes that had been collected thus far. If a match was found and the current website’s URL did not match the stored password hash’s URL, then a password re-use event was logged. The authors reported many interesting findings of users’ real-world password use, including the following highlights. Users had an average of 25 different online accounts, and typed 8 passwords on an average

day. Users maintained an average of 6.5 distinct passwords, each across 3.9 separate websites. Users predominately chose lowercase-only passwords unless required otherwise. Finally, based on their study’s results, the authors estimated that 0.4% of Internet users enter passwords on known phishing sites every year. Clearly, this study provided the research community with great insight into live user behavior, despite having only collected data for 3 months. However, unlike this study, we currently do not intend to collect data on participants’ passwords (see Section VI-B), given the risks (despite our security precautions) of storing such data for a study spanning several years.

De Luca et al. [11] observed 360 people’s interactions with automated teller machines (ATMs). A single experimenter personally monitored 60 people without their knowledge at each of 6 different banks’ ATMs at varied times of day. The goal of the study was to better understand the context of ATM usage without capturing users’ actual PINs. The data collected from each ATM interaction included the location, gender, time of day, interaction time, queue length, security measures taken by the user, and repeated PIN entry. The authors found that users were distracted in 11% of interactions, and that 65% of users made no effort to protect their PINs from observation attacks, either out of negligence, inability (e.g. carrying bags), or social context (e.g., did not want to imply mistrust in a nearby friend or family member). These and other results (including from interviews) led the authors to conclude that security should not rely on the user whenever possible, should be compatible with the social context, and PIN memorability is not a problem for most people, but it is severe when it occurs, since forgetting led to unsafe practices. The authors also shared lessons learned from the field observation study, including the utility of conducting pilot studies to test and refine the types and methods of data collection, abiding by strict codes of conduct to ensure ethical and consistent data collection, and importance of field studies in measuring users’ actual behavior, which can differ from users’ stated behavior in surveys and interviews.

VIII. CONCLUSION

Research to date has brought to light many usable security and privacy challenges computer users face, but there remain many unknowns, particularly with respect to home computer usages. Capturing data on these challenges in the wild as they occur naturally is essential if we are to conduct research and foster innovations with the greatest impact in improving the security and privacy of users and their machines. The Security Behavior Observatory (SBO) aims to collect said highly ecologically valid data on multiple security and privacy topics from hundreds of users’ home computers over several years. This paper has specified the SBO client-server architecture, the benefits of our design decisions, and the challenges and trade-offs involved in building a system with the reliability, robustness, and flexibility required for a study of this lengthy duration and grand scope. We hope the data collected will yield insights on a wide variety of security and

privacy challenges, and guide future research efforts towards solving the challenges users actually face in the wild.

REFERENCES

- [1] A. Adams and M. Sasse, "Users are not the enemy," *Communications of the ACM*, vol. 42, no. 12, 1999.
- [2] A. Whitten and J. Tygar, "Why Johnny can't encrypt: A usability evaluation of PGP 5.0," in *USENIX Security Symposium*, 1999.
- [3] R. Biddle, S. Chiasson, and P.C. van Oorschot, "Graphical passwords: Learning from the first twelve years," *ACM Computing Surveys*, vol. 44, no. 4, 2012.
- [4] A. Forget, "A world with many authentication schemes," Ph.D. dissertation, School of Computer Science, Carleton University, 2012.
- [5] C. Bravo-Lillo, L. Cranor, J. Downs, and S. Komanduri, "Bridging the gap in computer security warnings: A mental model approach," *Security & Privacy*, vol. 9, no. 2, 2011.
- [6] J. Hong, "The state of phishing attacks," *Communications of the ACM*, vol. 55, no. 1, 2012.
- [7] M. Jakobsson, "The human factor in phishing," *Privacy & Security of Consumer Information*, 2007.
- [8] I. T. L. Review, "G.r. newman and m.m. mcnelly," US Department of Justice, Tech. Rep. 210459, July 2005.
- [9] M. Brewer, "Research design and issues of validity," *Handbook of research methods in social and personality psychology*, pp. 3–16, 2000.
- [10] B. Berendt, O. Günther, and S. Spiekermann, "Privacy in e-commerce: Stated preferences vs. actual behavior," *Communications of the ACM*, vol. 48, no. 4, April 2005.
- [11] A. De Luca, M. Langheinrich, and H. Hussmann, "Towards understanding ATM security – a field study of real world ATM use," in *Symposium on Usable Privacy and Security (SOUPS)*. ACM, 2010.
- [12] D. Florêncio and C. Herley, "A large-scale study of WWW password habits," in *International World Wide Web Conference (WWW)*. ACM, May 2007.
- [13] M. Mazurek, S. Komanduri, T. Vidas, L. Bauer, N. Christin, L. Cranor, P. Kelley, R. Shay, and B. Ur, "Measuring password guessability for an entire university," in *Conference on Computer and Communications Security (CCS)*. ACM, 2012.
- [14] F. Lalonde Lévesque, J. Nsiempba, J. Fernandez, S. Chiasson, and A. Somayaji, "A clinical study of risk factors related to malware infections," in *Conference on Computer and Communications Security (CCS)*. ACM, 2013.
- [15] N. Christin, S. Egelman, T. Vidas, and J. Grossklags, "It's all about the benjamins: An empirical study on incentivizing users to ignore security advice," in *International Conference on Financial Cryptography and Data Security (FC)*. Springer, 2011.
- [16] D. Van Bruggen, S. Liu, M. Kajzer, A. Striegel, C. Crowell, and J. D'Arcy, "Modifying smartphone user locking behavior," in *Symposium on Usable Privacy and Security (SOUPS)*. ACM, 2013.
- [17] G. Friedland, G. Maier, R. Sommer, and N. Weaver, "Sherlock holmes' evil twin: On the impact of global inference for online privacy," in *New Security Paradigms Workshop (NSPW)*. ACM, 2011.
- [18] StatCounter.com, "Top 7 operating systems from july 2008 to nov 2013," October 2013, <http://gs.statcounter.com/#os-ww-monthly-200807-201311>.
- [19] —, "Mobile vs. desktop from july 2008 to oct 2013," accessed October 2013, http://gs.statcounter.com/#mobile_vs_desktop-ww-monthly-200807-201311.
- [20] S. Egelman, L. Cranor, and J. Hong, "You've been warned: an empirical study of the effectiveness of web browser phishing warnings," in *Conference on Human Factors in Computing Systems (CHI)*. ACM, 2008.
- [21] Microsoft Corporation, "Windows Installer (Windows)," November 2013, <http://msdn.microsoft.com/en-us/library/cc185688.aspx>.
- [22] —, "Services (Windows)," October 2013, <http://msdn.microsoft.com/en-us/library/windows/desktop/ms685141.aspx>.
- [23] —, "INFO: Run, RunOnce, RunServices, RunServicesOnce and Startup," November 2013, <http://support.microsoft.com/kb/179365>.
- [24] A. Menezes, P.C. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1996, ch. 10, p. 402, <http://cacr.uwaterloo.ca/hac/>.
- [25] S. Watanabe, *Solaris 10 ZFS Essentials*, 1st ed. Prentice Hall, 2010.
- [26] "OpenZFS," accessed November 2013, <http://open-zfs.org>.
- [27] T. White, *Hadoop: The Definitive Guide*, 3rd ed. O'Reilly, 2012.
- [28] Apache Software Foundation, "Welcome to apache hadoop," <https://hadoop.apache.org/>, accessed November 2013.
- [29] Federal Information Processing Standards (FIPS), "Advanced encryption standard," National Institute of Standards and Technology (NIST), Tech. Rep. 197, November 2001, <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>.
- [30] —, "Des modes of operations," National Institute of Standards and Technology (NIST), Tech. Rep. 81, December 1980, <http://www.itl.nist.gov/fipspubs/fip81.htm>.
- [31] R. Laboratories, "Pkcs #5: Password-based cryptography standard," accessed November 2013, <http://www.emc.com/emc-plus/rsa-labs/standards-initiatives/pkcs-5-password-based-cryptography-standard.htm>.
- [32] "Bson - binary json," accessed November 2013, <http://bsonspec.org/>.
- [33] A. Acquisti, "Privacy in electronic commerce and the economics of immediate gratification," in *Conference on Electronic Commerce*. ACM, 2004.
- [34] A. Shostack and P. Syverson, "What price privacy?" in *The Economics of Information Security*. Kluwer Academic Publishers, 2004.

PI: CRANOR

ARO Annual Progress Report

D) Scientific progress and accomplishments

I. STATEMENT OF THE PROBLEM STUDIED

So far, the research community has been relying on specific experiments or surveys to understand how users respond to specific security threats. Lab experiments and ad-hoc field experiments have offered insights into, for instance, how users fall to phishing or malware scams [1], [4], [5] or how users ignore security and privacy alerts [2], [6], [7]. These experiments offer a useful but narrow slice of user behavior under particular experimental settings. Developing scientific models of user behavior in response to security threats in a natural, real-world setting, is ultimately what we need to design sound security defenses.

Our ability to design appropriate information security mechanisms and sound security policies depends on our understanding of how end-users actually behave: End-users are not only the victims of cyber-threats, but also the passive participants in cyber-attacks originating from their own resources or exploiting their limitations. Models of users' behavior require extensive data collection; without that data, there cannot be a science of information security. To fill this gap, we are establishing the Security Behavior Observatory (SBO); a repository of data from a large panel of end-users whose online computing behavior will be captured, monitored, and analyzed over an extended period of time. This panel will offer an unprecedented window on real-time, real-life security and privacy behavior "in the wild." Through it, we aim at contributing to the evolution of a data-driven science of information security, with immediate applications in usability, economics, and secure system design.

The SBO will collect data about users' installation of and interactions with security products (e.g., anti-virus products, firewalls), their response to security-related alerts offered by browsers and operating systems, the websites they visits and the files they download that may expose them to malware and other security threats, as well as other security- and privacy-related behaviors. The SBO is currently targeting Windows computer users, but we plan to expand our methodology to other operating systems as well as mobile devices. Researchers will have opportunities to survey and interview panel members periodically to gain insights into why users behave in particular ways. In addition, researchers can use the observatory's infrastructure to not only collect data, but also to push stimuli (e.g., a simulated phishing email) and interventions (e.g., an alert about a new threat) out to panel members in order to collect data on subsequent changes in behavior. The data we will collect will provide a foundation on which sound models of user and attacker behavior. These models will eventually lead

to the scientific design of intervention policies and technical countermeasures against security threats.

The SBO has multiple scientific objectives. First, our panel will provide fertile grounds for multi-disciplinary research in computer science, human-computer interaction, behavioral sciences, and economics focused on understanding both end-user and attacker behaviors and strategies. This will lead to the creation of tools and products that can help users better protect themselves, and to tools and policies that better protect critical infrastructure. Thus, the SBO will not only lead to ground-breaking academic research, but it would also lead to actionable recommendations for policy makers and firms. Second, our work to develop the SBO provides insights into the design of sound measurement methodologies of end-user security behavior. Users' response to security and privacy threats remains largely unknown, in large part due to the significant difficulties in recruiting and monitoring users while addressing privacy concerns.

II. SUMMARY OF THE MOST IMPORTANT RESULTS

By the very nature of this project, which requires building infrastructure to collect data, then collecting, and eventually analyzing the data, there is a long setup phase. As a result, the project will be much more publication-centered toward the second half of its projected duration. However, we are confident that the more secure, reliable, and robust infrastructure as well as the greater number and quality of data collection sensors we have built and are refining will provide more and better data, resulting in more and stronger publications. Towards this end, we report the following accomplishments.

A. Dedicated Server Infrastructure

Designing and building the SBO requires attention to factors less frequently considered in shorter-term, more focused studies. The infrastructure must be sufficiently scalable, reliable, and robust to collect the required size, breadth, and depth of data over the study's lengthy duration. In addition, given the sheer amount and detail of behavioral data we will collect, we must carefully consider how best to maintain the security and privacy of participants' data while inconveniencing as little as possible researchers' working with the data. We also require the flexibility to adjust the types of data we collect throughout the study, since research needs will invariably change as earlier data analysis leads to further lines of inquiry.

Thus, we have designed, purchased, and deployed a dedicated server architecture that we believe best meets these requirements (see Figure 1). We summarize below each physical server's role, how data flows from the clients to the various

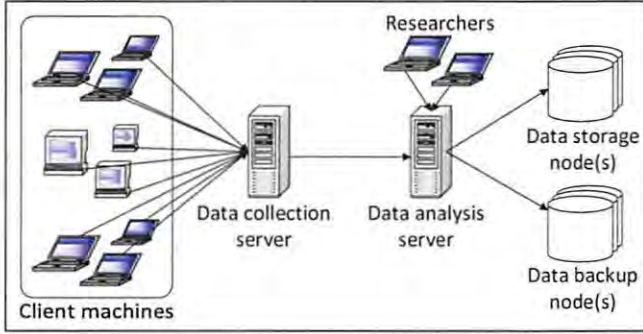


Fig. 1. Our SBO high-level hardware architecture and data flow.

server machines, and the security precautions that are in effect throughout. For more details on our client-server architecture and data security procedures, see our technical report [3].

1) *Data collection server*: The data collection server's role is solely to receive data from the panel's client machines, and periodically send said data to the data analysis server when requested. Data is encrypted before it is sent from the client to the data collection server (through an SSL secured tunnel). This server then stores the data locally, still encrypted with the client's encryption key.

2) *Data analysis server*: The data analysis server periodically retrieves the encrypted data from the data collection server (and thereafter deletes it from the data collection server), and stores it in the data storage node(s). To maintain data security, it must remain solely on the data analysis server and be accessible only to project administrators and researchers. Thus, the data analysis server can be accessed only through a secure shell (SSH) tunnel originating from the specific IP addresses of the researchers' and administrators' work machines. To remotely access the data analysis server, researchers and administrators must first remotely connect to their work machine and, through said machine, establish an SSH tunnel into the data analysis server. Since the data must never exist anywhere other than our servers, all work with the data must be performed through this SSH tunnel.

3) *Data node(s)*: Participants' encrypted data is ultimately stored in two places; in the data storage node(s) and data backup node(s). The backup node(s) are located in a physically-separate building from the storage nodes, both for security reasons and as a contingency for catastrophic events.

B. Data Collection Sensors

To best understand the nature of users' security and privacy challenges in computing, we must capture many types and depths of data. To capture such a wide array of data types over a long period of time, it is crucial we design and build an infrastructure that satisfies several requirements. First, we should minimize the impact of our data collection software on participants' computing and network performance. Thus, since the amount of data we can gather and transmit from clients is limited, we need the ability to be selective with and vary the types of data we collect over time. Second, as we

collect and analyze data, we expect our research questions will evolve and require different types of data to be answered. For these reasons, our data collection architecture must be flexible enough to accommodate our changing needs. Third, unlike most experimental software which is typically used for only a short time for specific targeted purposes and environments, any problems caused by our client software could profoundly impact participants' computing experience, due to the breadth, depth, and duration of our data collection. Thus, our system requires a much higher degree of stability and reliability than typical experimental software.

Thus, we have carefully designed our client data collection software to support many features necessary to provide us with the aforementioned requirements. These features are discussed in detail in our technical report [3]. One feature is support for data collection sensors that run independently of each other. This sensor independence provides the following robustness and adaptability benefits. Firstly, if a sensor fails, the other sensors will continue to collect data. Secondly, as the data interests for the study change over time, sensors can be silently and independently added, enabled, configured, disabled, or removed by the experimenters at any time without impacting any other aspect of the client system or our software. Finally, sensors can be implemented in whichever language is best for collecting the desired data.

We have been developing numerous sensors to collect many types of data, of which the following are ready for deployment in our pilot study (see Section II-C).

1) *Filesystem*: As currently designed, the SBO tracks changes to the filesystem, including the added, modified, or deleted file's size, last date modified, permissions, and other related information.¹ This data will help determine, for instance, if malware exists on the system and if so, how it affects machines' file systems, and whether or not users are likely to have noticed its presence.

2) *Installed software and operating system updates*: The SBO maintains a list of installed applications, their version numbers, and other related data, to determine what privacy or security software (e.g., anti-virus, firewall, ad-blockers, anonymizers) are installed, and whether they are up to date. The SBO also tracks which (and how soon after their release) operating system updates and patches have been installed. This allows us to measure the duration and severity of client machines' vulnerability to security threats.

3) *Processes*: The SBO monitors which processes (e.g., programs, applications) are running on clients' machines. It captures when all processes start and terminate, and can provide additional process status information at regular intervals. Primarily, this data will assist with the detection of malware. The SBO also collects general computer usage statistics that may help prioritize future security and privacy work, such as towards frequently-used applications.

4) *Security-related events*: The SBO also notes general security-related events, such as account-related events (e.g.,

¹However, we do not collect file or network packet contents since this may be too invasive and bandwidth intensive.

logins, settings changes, password changes), registry modifications, wireless network authentications, firewall changes, and potential attacks detected by the operating system. This will provide valuable insights on multiple usable security topics, including the security measures users' employ on their computers, potentially dangerous program behavior, and the types and frequency of attacks that occur on home users' machines.

5) *Network traffic*: The SBO captures all network packet headers sent and received to clients' computers.¹ This data would allow us to detect various network traffic types that may be risky (e.g., peer-to-peer file transfers, dangerous websites) or suspicious (e.g., malware, intrusion attacks). We could thereby verify whether risky Internet behavior is correlated with a higher probability of an attack or infection.

6) *Internet browsing behavior*: We intend to further monitor users' web browsing behavior by collecting data from Microsoft Internet Explorer, Mozilla Firefox, and Google Chrome. We intend to capture search queries, online social network activity, browsers' and some online accounts' privacy and security settings, as well as other behavior of particular research interest (e.g., social networks, behavioral advertising). One example of possible analyses includes: what are users' privacy settings and behaviors on online social networks, do said settings adequately preserve users privacy, and if not, how could the website be better designed to empower users to more easily and accurately express their desired privacy settings. Another example of planned analysis consists of measuring how often users' actually make purchases derived from behavioral advertising links. This would reveal insights on the actual utility users gain from behavioral advertising, with respect to the privacy cost.

C. Pilot Study

Now that we have a secure, reliable, and scalable dedicated server infrastructure (see Section II-A) on which to store and manage the massive amount of data the SBO clients' data sensors (see Section II-B) will collect, we are now launching a pilot study with a small number of participants from the general population to test our data collection infrastructure. The pilot study will be the ideal beta test for our SBO study participant recruitment and enrollment process, as well as for our data collection systems to ensure they will have a minimal impact on the performance of clients' machines while providing us with sample data to test our data collection and analysis methodologies. While we cannot predict the number and severity of problems that may arise, we hope that there will be few such problems and that we will be ready to begin full data collection from 50-100 client machines soon. In the meantime, we hope to compile the lessons learned about building and launching such a large-scale field study into an early publication. We also hope the pilot will go smoothly enough that we could submit a paper with early results from the short-term data collected.

REFERENCES

- [1] N. Christin, S. Egelman, T. Vidas, and J. Grossklags. It's all about the benjamins: An empirical study on incentivizing users to ignore security advice. In *International Conference on Financial Cryptography and Data Security (FC)*. Springer, 2011.
- [2] S. Egelman, L. Cranor, and J. Hong. You've been warned: an empirical study of the effectiveness of web browser phishing warnings. In *Conference on Human Factors in Computing Systems (CHI)*. ACM, 2008.
- [3] A. Forget, S. Komanduri, A. Acquisti, N. Christin, L. Cranor, and R. Telang. Security behavior observatory: Infrastructure for long-term monitoring of client machines. Technical Report CMU-CyLab-14-009, CyLab, Carnegie Mellon University, 2014.
- [4] S. Y. N. Christin and K. Kamataki. Dissecting one click frauds. In *Conference on Computer and Communications Security (CCS)*. ACM, 2010.
- [5] S. Sheng, M. Holbrook, P. Kumaraguru, L. Cranor, and J. Downs. Who falls for phish? a demographic analysis of phishing susceptibility and effectiveness of interventions. In *Conference on Human Factors in Computing Systems (CHI)*. ACM, 2010.
- [6] J. Sunshine, S. Egelman, H. Almuhiemedi, N. Atri, and L. Cranor. Crying wolf: an empirical study of ssl warning effectiveness. In *USENIX Security Symposium*, 2009.
- [7] M. Wu, R. Miller, and S. Garfinkel. Do security toolbars actually prevent phishing attacks? ACM, 2006.

Composability of Big Data and Algorithms for Social Networks Analysis Metrics

PI(s): Jürgen Pfeffer

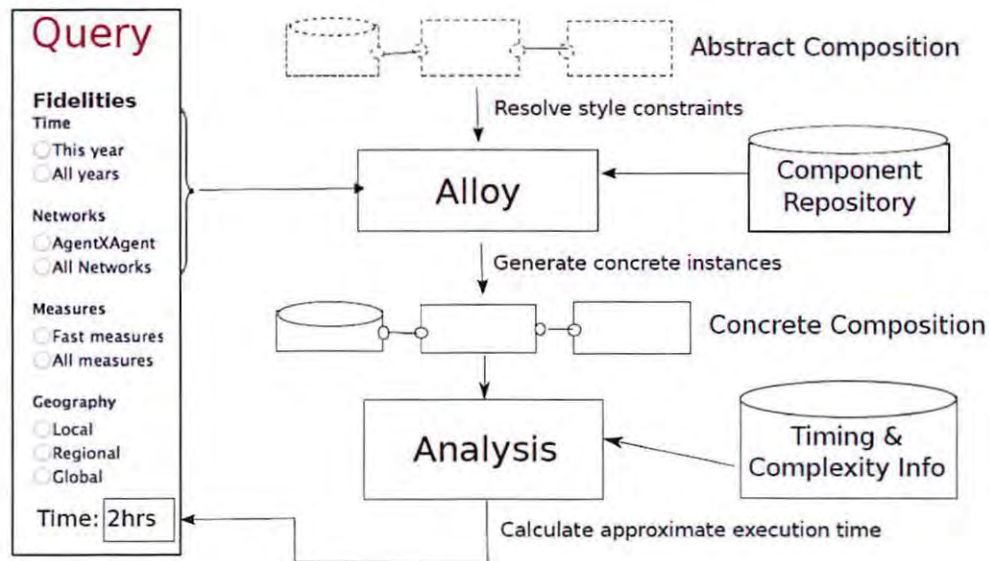
Scientific progress and accomplishments

- Although, network metrics are not necessarily robust for sampled data, we could show that estimating the goodness of fit of the metrics estimation is possible. Non-linear fit is superior to linear fit.

Nonlinear least squares (NLS): $r(C_y, C_y^S) = \beta_0 + \beta_1 n + \beta_2 e + [\beta_3 n_s^{\alpha_1}] + \beta_4 (1 - e_s^{\alpha_2})$

Covariate	Description	Predictor	Degree Centrality		Betweenness Centrality	
			Email	Message	Email	Message
		Intercept	- 1.305 ***	- 0.354 **	- 0.653 ***	0.230 ***
n	original network size, $ V $	n	- 0.146 ***	- 0.323 ***	- 0.226 ***	- 0.300 ***
e	original network edge communication count, $ E $	e	- 0.504 *	33.133 ***	1.613 ***	30.414 ***
n_s	sampled node count	$n_s^{\alpha_1}$	0.000	0.136 ***		
e_s	sampled edge communication count	α_1	2.296 ***	0.490 ***		
		$1 - e_s^{\alpha_2}$	2.587 ***	1.249 ***	3.184 ***	2.556 ***
		α_2	0.359 ***	0.560 ***	0.107 ***	0.056 ***
		AIC	- 16970	- 29993	- 11766	- 19219
		BIC	- 16918	- 29937	- 11727	- 19177
		Adj-R ²	0.783	0.736	0.794	0.631

- In collaboration with David Garlan, Bradley Schmerl, and Vishal Dwivedi (Ph.D. student) we studied the integration of metrics optimization in software architectures. Results of this collaboration are a publication (Dwivedi et al, 2014) as well as a new project for the 2014+ Lablet.



- The PI ran a “big data” session at the last Sunbelt 2013 conference for social network analysis. In our presentation we introduced the metrics optimization problem from a software architecture perspective including several layers for possible optimizations.

